Electronic Colloquium on Computational Complexity, Report No. 159 (2012)

A Combinatorial Characterization of smooth LTCs and Applications

Eli Ben-Sasson* Michael Viderman†
Computer Science Department
Technion — Israel Institute of Technology
Haifa, 32000, Israel.
{eli, viderman}@cs.technion.ac.il

November 19, 2012

Abstract

The study of locally testable codes (LTCs) has benefited from a number of nontrivial constructions discovered in recent years. Yet we still lack a good understanding of what makes a linear error correcting code locally testable and as a result we do not know what is the rate-limit of LTCs and whether asymptotically good linear LTCs with constant query complexity exist.

In this paper we provide a combinatorial characterization of smooth locally testable codes, which are locally testable codes whose associated tester queries every bit of the tested word with equal probability. Our main contribution is a combinatorial property defined on the Tanner graph associated with the code tester ("well-structured tester"). We show that a family of codes is smoothly locally testable if and only if it has a well-structured tester.

As a case study we show that the standard tester for the Hadamard code is "well-structured", giving an alternative proof of the local testability of the Hadamard code, originally proved by [Blum, Luby, Rubinfeld, STOC 1990]. Additional connections to the works of [Ben-Sasson, Harsha, Raskhodnikova, SICOMP 2005] and of [Lachish, Newman and Shapira, Computational Complexity 2008] are also discussed.

*The research has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement number 240258. Part of work done while visiting the Computer Science and Artificial Intelligence Laboratory (CSAIL) of MIT, Cambridge, MA, USA.

†The research has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement number 202405.

ISSN 1433-8092

Contents

1 Introduction

Locally testable codes (LTCs) are error correcting codes for which membership in the code can be tested to a high degree of certainty via a randomized testing procedure that reads only a few bits from the purported codeword. This testing procedure accepts all legal codewords, and rejects with non-negligible probability all words that are sufficiently far from every codeword in Hamming distance. Certain LTCs (with additional properties) are a crucial building block in the construction of probabilistically checkable proofs (PCPs) [6, 5] and as a result LTCs were defined and studied in early works on PCPs [42, 4, 24, 41]. A more systematic study of LTCs was initiated by [25] and since then a number of nontrivial constructions of families of LTCs have been discovered, including LTCs based on PCPs of proximity (PCPPs) [10, 22, 37], on tensors of codes [16, 17, 18, 21, 23, 44], on sparse codes [30, 32, 19] and on affine-invariant codes [11, 15, 31, 29].

In spite of this impressive progress, a number of basic questions regarding LTCs remain unsolved. First and foremost is the question of the existence of a family of asymptotically good LTCs, i.e., a family of codes of arbitrarily large blocklength that all have constant (non-zero) rate and are locally testable with a constant number of queries (cf. [9]). One of the major problems impeding progress on this fundamental question is our lack of understanding of what makes a code locally testable, and this is true even for the special case of linear codes that is nevertheless quite interesting (most known LTCs are linear). Recall that an $[n, k, d]_{\mathbb{F}}$-code C is a k-dimensional linear subspace of $\mathbb{F}^n$ where $\mathbb{F}$ is a finite field and every nonzero codeword $w \in C$ has at least d nonzero entries. Soon we will also rely on the notion of the dual code $C^{\perp}$, which is the linear space that is dual to C in $\mathbb{F}^n$.

It was shown by [13] that for a linear code to be testable with q queries it must be the case that the distance of its dual code is at most q, yet this by itself is far from being a sufficient condition for local testability. Indeed, they showed a random low-density-parity-check (LDPC) code is not locally testable even though C is characterized as the set of codewords that satisfy all low-density-parity checks. Furthermore, any LTC must have a certain amount of "redundant" low-weight dual words that are nontrivial combinations of the constraints that characterize C [12]. Somewhat surprisingly, even when a code is characterized by a large pair-wise independent set of low-weight constraints, as is the case for affine-invariant codes, this is not sufficient for guaranteeing local testability [14]. This brief survey highlights the mystery surrounding LTCs and calls for a better understanding of the combinatorial and algebraic characterization of LTCs.

In this paper we provide such a characterization for *smooth* linear LTCs. A smooth LTC is one in which the associated tester queries each coordinate with roughly the same probability, i.e., the distribution on $\{1, \ldots, n\}$ induced by selecting a random bit in a random query of the tester is roughly uniform (see Definition 2.1). Our characterization says that a family of codes is locally testable with query complexity q if and only if one can construct a Tanner graph with certain properties, explained next for the simple case of binary codes which are codes over the two-element field $\mathbb{F}_2$.

Code testers as Tanner graphs Every binary linear code C can be characterized by a bipartite Tanner graph G with left vertex set $\{1, \ldots, n\}$ and right vertex set U as:

$$C = \left\{ w \in \mathbb{F}_2^n \;\middle|\; \forall u \in U, \sum_{i \in N(u)} w_i = 0 \right\}$$

where w_i denotes the ith entry of w and $N(u)$ is the set of neighbors of u in G. [1] There are many Tanner graphs that characterize C, one graph for each set of vectors $U = \{u^{(1)}, \ldots, u^{(m)}\}$ that span $C^\perp$, where we set the neighborhood of (the vertex) $u^{(j)}$ in G to be the support of (the vector) $u^{(j)}$, i.e., its set of nonzero entries denoted as $\mathrm{supp}(u^{(j)}) = \left\{ i \;\middle|\; u_i^{(j)} \neq 0 \right\}$. Due to the correspondence between right vertices in G and dual codewords of C we will often refer to right vertices as dual codewords, or "constraints" imposed on C. Now, since any q-query tester for C is defined by a distribution $\mathcal{D}$ over words in $C^\perp$ of support size at most q (cf. [12, Section 2]), one can equate a q-query tester for C with a Tanner-graph G of right-degree at most q, where the probability of picking a dual word u under $\mathcal{D}$ is proportional to the number of vertices whose set of neighbors is $\mathrm{supp}(u)$.

A nontrivial result of [12] says that not all Tanner graphs of an LTC correspond to good testers. For instance, Tanner graphs corresponding to a basis for $C^\perp$ are invariably bad. But based on our previous discussion we know that every LTC must have some Tanner graph that corresponds to a good tester and the main result of this paper is to characterize LTCs in terms of their Tanner graphs.

Characterization of smooth linear LTCs in terms of their Tanner graphs Informally speaking, we show that a code is smoothly locally testable with query complexity q if and only if it has a "locally redundant" and "stable" Tanner graph of right-degree at most q. A *locally redundant* Tanner graph has the property that for any constraint (right vertex) u with neighborhood $I = \mathrm{supp}(u)$, if an adversary removes an ϵ fraction of the constraints that touch I, then the remaining constraints still span u. In other words, local redundancy means that every constraint u in the Tanner graph is spanned "locally" by other constraints that touch $\mathrm{supp}(u)$, and moreover this property is redundant, so much so that even when a non-negligible fraction of u's neighboring constraints are removed, one can still recover u as a linear combination of its remaining local neighbors. A Tanner graph is called *stable* if it has the property that removing all constraints that touch a small set of indices $I \subset [n]$, leaves one with a Tanner graph that still pretty much characterizes the same code C (up to discarding a small auxiliary corrupted set of bits I').

Our main result, Theorem 2.7, shows that a family of codes is smoothly locally testable if and only if every code in the family has a Tanner graph subscribing to the previous two requirements. The forward direction, going from a smoothly-testable code to a locally redundant and stable Tanner graph is fairly straightforward and it is the backward direction that is nontrivial. To prove it, i.e., to show that a Tanner graph that is locally redundant and stable implies a locally testable code, we argue that any word w that is far from the code contains a large fraction of bits that are each involved in many constraints that are unsatisfied by w. Therefore, sampling a random constraint from the Tanner graph and using it as a test will reject w with constant probability.

To show that our characterization is meaningful we use it to show a new proof of the celebrated result of [20] showing that the family of Hadamard codes is locally testable, by arguing that the Tanner graph corresponding to the BLR-tester is locally redundant and stable. While the bounds obtained from our method (cf. Theorem 2.9) are weaker than those arising from the Fourier-based analysis of [8, 28], we think that obtaining a different proof for the most basic and well-studied locally testable code is of value.

[1] Tanner graphs of codes over larger fields must have weighted edges, a complication that we choose to avoid in this introductory discussion.

1.1 A relation to the work of Ben-Sasson et al. [13]

Ben-Sasson et al. [13] showed that to test some 3-CNF formulas it is necessary to use $\Omega(n)$ queries to the input assignment to check whether it is close to satisfy the formula. This result was obtained by showing that if a Tanner graph of a regular LDPC code $C \subseteq \mathbb{F}_2^n$ has certain expansion properties then to test the code requires $\Omega(n)$ queries. These expansion properties required two assumptions.

The first assumption is that any small subset of left vertex set has many unique neighbors on the right side of the underlying Tanner graph, or equivalently, every small subset of indexes has many unique neighbor constraints of small weight. Assume now that the codeword indices are labeled by elements of the cyclic group of size n. Let us say informally that the cyclic-width of a constraint is large if the maximum distance between two coordinates in its support is $\Omega(n)$, and otherwise we say the cyclic-width of the constraint is small (see Section 3 for a formal definition of cyclic-width). We will show that the first assumption implies that the cyclic-width of the constraints of S must be large, i.e. (see Claim 3.1). The second assumption is that every small weight constraint can be obtained only by a linear combination of a small number of other small weight constraints (from the right side of the Tanner graph).

Roughly speaking, using our characterization of smooth LTCs (in particular, Lemma 2.11) we show a result that contrasts with that of [13]. Namely, we show that if the first assumption does not hold but the second one does, then the code is locally testable. In different words, a code that has *small cyclic-width* (negation of first assumption) but each of its constraint u can be expressed as a sum of a small number of "adjacent" constraints (second assumption), is locally testable. See Section 3 for more details.

1.2 Space complexity vs. Query complexity

One of the research lines in property testing investigates the testability of low complexity languages. Alon et al. [1] proved that all regular languages are testable with a constant number of queries. It has been long known (see Exercise 2.8.12 in [39]) that $\mathbf{DSPACE}(o(\log\log n))$ is exactly the set of regular languages. Hence, the work of [1] implies that all languages in $\mathbf{DSPACE}(o(\log\log n))$ are testable with $O(1)$ of queries.

This result was generalized later by Newman [38] who investigated the relation between space complexity and query complexity. He proved that languages that have constant cyclic-width branching program are testable with constant number of queries. Newman raised a question about the maximal gap between the space complexity and the query complexity. Lachish et al. [33] continued to study the relation between space complexity and query complexity of languages. They showed that for any space constructible function $s(n)$[2], there exists a language $L \in \mathbf{DSPACE}(s(n))$ whose testing requires $2^{\Omega(s(n))}$ queries. Lachish et al. conjectured that their result is tight, i.e., the query complexity of any language is at most exponential in its space complexity.

Conjecture 1.1 ([33])**.** *For any function $s(n)$ and language $L \in \mathbf{DSPACE}(s(n))$ it holds that L is testable with $2^{O(s(n))}$ queries.*

Clearly, the relevant range for $s(n)$ is $s(n) \leq O(\log n)$ since any decidable language is testable

[2]The notion of "space constructibility" they used is the standard one in the complexity theory, see e.g. [39]. A function $s(n)$ is called *space-constructible* if there exists a Turing machine M which, given a string 1^n consisting of n ones, outputs the binary representation of $s(n)$, while using only $O(s(n))$ space.

with n queries.[3] As was explained above, this conjecture was known to be true for $s(n) = o(\log\log n)$ due to [1]. In this way, Conjecture 1.1 remains open for $\Omega(\log\log n) \leq s(n) \leq O(\log n)$. This range of the space function is usually referred to as sublogarithmic space computations.

In spite of the beautiful research on limitations of the computations with sublogarithmic space, e.g. [2, 3, 26, 27, 34], and the progress on understanding of the power and limitations of sublogarithmic space computations, it seems that it is hard to resolve Conjecture 1.1 in the current state. We suggest to simplify the task and consider this conjecture for linear languages $L = \bigcup_n L_n$ satisfying that for every $n \in \mathcal{N}$ it holds that $L_n \subseteq \mathbb{F}_2^n$ is a linear subspace of $\mathbb{F}_2^n$. In other words, we suggest to try and prove the following corollary of Conjecture 1.1.

Conjecture 1.2. *For any function $s(n)$ and a linear language $L \in \mathbf{DSPACE}(s(n))$ it holds that L is testable with $2^{O(s(n))}$ queries.*

In Section 3.3 we address Conjecture 1.2 and describe some progress towards resolving it. Our proofs use the observations related to the structured codes and the fact that structured codes imply local testability (Lemma 2.11).

Organization of the paper. In the following section we provide the standard definitions regarding locally testable codes and state our main results. In particular, Section 2.3 contains a definition of structured codes and Section 2.4 contains the statements of our main results. In Section 3 we discuss connections to the works [13, 33] and state Theorems 3.3 and 3.8. Sections 4 and 5 show that structured codes are (smooth) LTCs and that smooth LTCs are structured. In Section 6 we show that Hadamard codes are strongly structured. Finally, in Sections 7 and 8 we prove Theorems 3.3 and 3.8, respectively.

2 Definitions and Main Results

After presenting the standard definition of locally testable codes we introduce the notion of a "structured" code, then prove that a family of linear codes is a smooth LTC if and only if it is structured. We start with necessary notation.

2.1 Notation

Let $[n]$ be the set $\{1, \ldots, n\}$. We assume that $i \mod n$ is an element of $[n]$. In particular, $n \mod n = n$ and $(n+1) \mod n = 1$. For a finite set $A \subset \mathcal{N}$ we let $(A \mod n) = \{a \mod n \mid a \in A\}$.

$\mathbb{F}$ will invariably denote a finite field. A linear code over $\mathbb{F}$ is a linear subspace $C \subseteq \mathbb{F}^n$. The dimension of C, denoted by $\dim(C)$, is its dimension as a vector space. The rate of C is the ratio of its dimension to n. We define the distance between two words $x, y \in \mathbb{F}^n$ to be $\Delta(x, y) = |\{i \mid x_i \neq y_i\}|$ and the relative distance to be $\delta(x, y) = \frac{\Delta(x,y)}{n}$. The distance of C is defined by $\Delta(C) = \min_{x \neq y \in C} \Delta(x, y)$ and its relative distance is defined by $\delta(C) = \frac{\Delta(C)}{n}$. We note that $\Delta(C) = \min_{c \in C \setminus \{0\}} \{\mathrm{wt}(c)\}$.

[3] If L is a decidable language then there exists a Turing Machine M, which decides it. By definition, a tester for L has no running time restrictions. Thus, the tester can query all bits from the input word $w \in \{0,1\}^n$, and after that to run M on w to decide whether it belongs to the language.

For $w \in \mathbb{F}^n$, let $\mathrm{supp}(w) = \{i \in [n] \mid w_i \neq 0\}$ and $|w| = |\mathrm{supp}(w)|$. For $x \in \mathbb{F}^n$ and $C \subseteq \mathbb{F}^n$, let $\delta(x, C) = \min_{y \in C}\{\delta(x, y)\}$ denote the relative hamming distance of x from the code C. If $\delta(x, C) \geq \epsilon$, we say that x is ϵ-far from C and otherwise x is ϵ-close to C. For $u = (u_1, u_2, \ldots, u_n), v = (v_1, v_2, \ldots, v_n) \in \mathbb{F}^n$ let $\langle u, v\rangle$ denote the bilinear function from $\mathbb{F}^n \times \mathbb{F}^n$ to $\mathbb{F}$ defined by $\langle u, v\rangle = \sum_{i=1}^{n} u_i v_i$. For $T \subseteq \mathbb{F}^n$ we say that $w \perp T$ if for all $t \in T$ we have $\langle w, t\rangle = 0$. The dual code $C^\perp$ is defined as $C^\perp = \{u \in \mathbb{F}^n \mid u \perp C\}$. Similarly, we define $C^\perp_{\leq t} = \left\{u \in C^\perp \mid |u| \leq t\right\}$ and $C^\perp_t = \left\{u \in C^\perp \mid |u| = t\right\}$. For $w \in \mathbb{F}^n$ and $S = \{j_1, j_2, \ldots, j_m\} \subseteq [n]$ we let $w|_S = (w_{j_1}, w_{j_2}, \ldots, w_{j_m})$, where $j_1 < j_2 < \ldots < j_m$, be the restriction of w to the subset S. Similarly, we let $C|_S = \{c|_S \mid c \in C\}$ denote the projection of the code C onto S. For $A \subseteq C^\perp$ and $J \subseteq [n]$ we let $A_{(-J)} = \{u \in A \mid \mathrm{supp}(u) \cap J = \emptyset\}$.

2.2 Families of linear locally testable codes

We extend the standard definition of LTCs (cf. [12]) to families of linear codes. In this paper, a *family of codes* refers to an infinite sequence $\{C_{n_i}\}_{i=1,2,\ldots}$ of $[n_i, k_i, d_i]_{\mathbb{F}}$-linear codes C_i satisfying $n_1 < n_2 < \ldots$ and all having relative distance at least δ for some $\delta > 0$.

Definition 2.1 (Families of LTCs)**.** A *q-query tester*, or simply *q-tester*, for a linear code $C \subseteq \mathbb{F}^n$ is a distribution $\mathcal{D}$ supported on $S \subseteq C^\perp_{\leq q}$. $\mathcal{D}$ is said to be

A (q, ϵ, δ)-tester if for all $w \in \mathbb{F}^n$, $\delta(w, C) \geq \delta$ we have $\Pr_{u \sim \mathcal{D}}[\langle u, w\rangle \neq 0] \geq \epsilon$.

A (q, ϵ)-strong tester if for all $w \in \mathbb{F}^n$ we have $\Pr_{u \sim \mathcal{D}}[\langle u, w\rangle \neq 0] \geq \epsilon \cdot \delta(w, C)$.

t-smooth if for all $i \in [n]$ we have $\Pr_{u \sim \mathcal{D}}[i \in \mathrm{supp}(u)] \leq t/n$.

Accordingly, $C \subseteq \mathbb{F}^n$ is a *(q, ϵ, δ)-LTC* (*(q, ϵ)-strong LTC*, respectively) if it has a (q, ϵ, δ)-tester ((q, ϵ)-strong tester, respectively). If the associated tester is t-smooth we say C is a *(q, ϵ, δ, t)-smooth LTC* or a *(q, ϵ, t)-smooth strong LTC*, respectively.

Let $\mathcal{F} = \{C_{n_i}\}_{i=1,2,\ldots}$ be a family of linear codes with relative distance δ such that each member has a q-tester $\mathcal{D}_i$. $\mathcal{F}$ is called a q-LTC if for every $\delta' \in (0, \delta/3)$ there exist i_0 and $\epsilon > 0$ such that for $i > i_0$, the tester $\mathcal{D}_i$ is a (q, ϵ, δ')-tester. If every $\mathcal{D}_i$ is (q, ϵ)-strong for some $\epsilon > 0$, then F is a q-strong LTC. Finally, if there exists t such that every $\mathcal{D}_i$ is t-smooth we say that the family is smooth.

2.3 Families of structured codes

The notion of structured codes is the main contribution of this paper, because it will turn out that families of codes are smooth and structured if and only if they are smooth and locally testable.

Before presenting the definition we give an intuitive graph-based explanation for it. For $C \subset \mathbb{F}^n$ a linear code, $J \subset [n]$, and $S \subset C^\perp$ let

$$N_S(J) = \{u \in S \mid \mathrm{supp}(u) \cap J \neq \emptyset\}.$$

For $v \in \mathbb{F}^n$ we let $N_S(v) = N_S(\mathrm{supp}(v))$, noting that if u is a nonzero element of S then $u \in N_S(u)$. For $i \in [n]$ we simplify notation and set $N_S(i) = N_S(\{i\})$.

Tanner graph intuition Informally, a q-tester for a linear code $C \subseteq \mathbb{F}^n$ can be described by a distribution over (right) vertices in a Tanner-graph [43]. Given $S \subseteq C^\perp$ that spans $C^\perp$, the *Tanner graph* $([n], S, E, e)$ corresponding to S consists of a bipartite graph $G([n], S, E)$, where $E = \{(i, u) \mid i \in [n], u \in S, i \in \mathrm{supp}(u)\}$ and a function $e : E \to \mathbb{F} \setminus \{0\}$. This Tanner graph defines the code $C \subseteq \mathbb{F}^n$ via the rule that for all $x \in \mathbb{F}^n$ it holds that $x \in C$ if and only if for all $u \in S$ we have $\sum_{j \in N(u)} x_j \cdot e(j, u) = 0$, where $N(u)$ denotes the set of neighbors of u in the graph.

Therefore, given a q-tester $\mathcal{D}$ supported on $S \subset {C^\perp}_{\leq q}$ the resulting Tanner graph will have right-degree at most q and $\mathcal{D}$ defines a distribution over the right vertices of G.

For a code to be structured, it should have a Tanner graph that fulfills two requirements. The first is a locality property that says, informally, that every constraint $u \in S$ is redundantly spanned by $N_S(u)$, i.e., even if an ϵ-fraction of $N_S(u)$ is removed, u is still spanned by the remaining constraints in its neighborhood.

The second requirement is of a global nature, and says, informally, that even when an ρ-fraction of coordinates $J \subset [n]$ are removed, then one can throw a small additional fraction ρ' of coordinates $J' \subset [n]$ such that the remaining constraints in S that are supported on $[n] \setminus J$ span the code that is the projection of C onto $[n] \setminus (J \cup J')$. Crucially, we will require ρ' to approach 0 as ρ goes to 0. We now give the formal definitions. In what follows, for $\mathcal{D}$ a distribution supported on a set S and $S' \subset S$ let $\mathcal{D}(S') = \sum_{s \in S'} \mathcal{D}(s)$.

Definition 2.2 (Local Redundancy)**.** Let $C \subseteq \mathbb{F}^n$ be a linear code and $\mathcal{D}$ be a distribution supported on $S \subseteq C^\perp$. For $\epsilon \in (0, 1)$ we say that C is ϵ-redundant with respect to S and $\mathcal{D}$ if for any $u \in S$ and $S' \subseteq S$ such that for every $i \in \mathrm{supp}(u)$ we have $\mathcal{D}(N_S(i) \cap S') < \epsilon \cdot \mathcal{D}(N_S(i))$, it holds that $u \in \mathrm{span}(S \setminus S')$.

Remark 2.3. We stress that every tester is ϵ-redundant for some $\epsilon > 0$, by taking $\epsilon = \min_{u \in S} \mathcal{D}(u)$. For such small ϵ, inspecting the previous definition we see that the only possible S' there is the empty set. This trivially implies $u \in \mathrm{span}(S \setminus S') = \mathrm{span}(S)$ because $u \in S$.

We say that $S \subseteq C^\perp$ characterizes C if $\mathrm{span}(S) = C^\perp$. In this case, for all $x \in \mathbb{F}^n$ we have: $x \in C$ if and only if $x \perp S$. Intuitively, we say that C is *stable* with respect to S if even after some vectors are removed from S it still approximately characterizes the code C.

Definition 2.4 (Stable Characterization)**.** Let $0 < \rho' \leq 1$ and $S \subseteq C^\perp$. We say that C is ρ'-(strictly) characterized by S if there exists $J' \subseteq [n]$ such that $|J'| \leq \rho' n$ ($|J'| < \rho' n$) and $(C^\perp)_{(-J')} \subseteq \mathrm{span}(S)$.

We say that C is (ρ, ρ')-stable with respect to S if for all $J \subset [n]$, $|J| < \rho n$ it holds that C is ρ'-strictly characterized by $S_{(-J)}$. We say that C is (ρ, ρ')-strongly stable with respect to S if for all $J \subset [n]$, $|J| < \rho n$ it holds that C is $\min\left\{\left(\frac{|J|}{n} \cdot \frac{1}{\rho'}\right), 1\right\}$-characterized by $S_{(-J)}$.

We are ready to introduce our main definition, that of families of structured codes.

Definition 2.5 (Families of structured codes)**.** A linear code $C \subseteq \mathbb{F}^n$ is a $(q, \epsilon, \rho, \rho')$-structured (strongly structured) if there exists a distribution $\mathcal{D}$ supported on $S \subseteq {C^\perp}_{\leq q}$ such that the following three conditions hold.

Local redundancy C is ϵ-redundant with respect to S and $\mathcal{D}$,

Stability C is (ρ, ρ')-stable (strongly stable) with respect to S,

Sampling For all $I \subseteq [n]$, $|I| \geq \rho n$ ($|I| \geq 0$) such that $\forall i \in I, \exists u \in S : i \in \mathrm{supp}(u)$ we have

$$\Pr_{u \sim \mathcal{D}}[\mathrm{supp}(u) \cap I \neq \emptyset] \geq \epsilon \cdot \frac{|I|}{n}.$$

Let $\mathcal{F} = \{C_{n_i}\}_{i=1,2,\ldots}$ be a family of linear codes such that each member has a q-tester $\mathcal{D}_i$. $\mathcal{F}$ is said to be q-structured if for every $\rho \in (0,1)$ there exists $\rho' = \rho'(\rho)$, $\epsilon = \epsilon(\rho)$ and i_0 such that both conditions hold:

1. Both ρ' and ϵ go to 0 as ρ goes to 0.
2. For every $i > i_0$ the code C_i is $(q, \epsilon, \rho, \rho')$-structured.

If, additionally, there exists t such that every $\mathcal{D}_i$ is t-smooth we say that $\mathcal{F}$ is a smooth q-structured family.

Remark 2.6. Note that local redundancy and sampling property were defined using the same parameter ϵ. This is done for the sake of convenience, since otherwise one could use ϵ_1 for local redundancy, ϵ_2 for sampling and then set $\epsilon = \min\{\epsilon_1, \epsilon_2\}$.

Our main Theorem is the following, its proof appears in Section 2.4.

Theorem 2.7 (Main Theorem)**.** *A family $\mathcal{F}$ of linear codes is smooth q-LTC if and only if it is smooth q-structured.*

As a proof of concept we provide a simple combinatorial proof that the family of Hadamard codes, defined next, is strongly structured, and this implies, via Lemma 2.11 that it is also strong locally testable, thereby recovering the proof of the famous linearity-testing theorem of [20].

Definition 2.8 (The Hadamard family)**.** For integer k let $G_k \in \mathbb{F}_2^{(2^k-1)\times k}$ be the matrix whose rows are all distinct nonzero vectors in $\mathbb{F}_2^k$. Then $C_k = \{G_k \cdot x \mid x \in \mathbb{F}_2^k\}$ is the Hadamard code of dimension k and the Hadamard family of codes is $\{C_k\}_{k=1,2,\ldots}$.

Theorem 2.9 (Hadamard codes are strongly structured)**.** *A Hadamard code $C \subseteq \mathbb{F}_2^n$ is $(3, \frac{1}{3}, \frac{1}{6}, 1)$-strongly structured and 3-smooth. Hence, Theorem 2.7 implies that C is a smooth q-LTC. Moreover, Lemma 2.11 below implies that C is $(3, \frac{1}{54})$-strong LTC and 3-smooth.*

The proof of Theorem 2.9 is postponed to Section 6. We end this section by giving a simple and easy-to-analyze example of a structured family of codes.

Example 2.10 (Repetition code)**.** Let $C = \{0^n, 1^n\}$ and let $\mathcal{D}$ be the uniform distribution over $S = C_2^\perp = \{u \mid |u| = 2\}$.

First of all, one can see that C is $\frac{1}{2}$-redundant with respect to S and $\mathcal{D}$. This fact follows from the observation that for $u \in S$ all its neighbors can be written as a list of disjoint pairs $(u_1, u_1'), (u_2, u_2'), \ldots$ such that for every i we have $u_i + u_i' = u$. So, if $S' \subseteq S$ such that for every $i \in \mathrm{supp}(u)$ we have $\mathcal{D}(N_S(i) \cap S') < \epsilon \cdot \mathcal{D}(N_S(i))$, then $u = u_i + u_i'$ for some $u_i, u_i' \in \mathrm{span}(S \setminus S')$.

It can also be readily verified that C is $(1,1)$-strongly stable. To see this let $I \subseteq [n]$ be a subset and $S' = \{u \in S \mid \mathrm{supp}(u) \cap I = \emptyset\}$ then $S \setminus S'$ characterizes the code $C|_{[n]\setminus I}$, i.e., $(C^\perp)_{(-I)} \subseteq \mathrm{span}(S \setminus S')$. Finally, for all $I \subseteq [n]$ we have $\Pr_{u \sim \mathcal{D}}[\mathrm{supp}(u) \cap I \neq \emptyset] \geq \frac{1}{2} \cdot \frac{|I|}{n}$.

Thus C is $(2, \frac{1}{2}, 1, 1)$-strongly structured (Definition 2.5) and hence by Lemma 2.11, C is a $(2, \frac{1}{2^2})$-strong LTC.

2.4 Proof of Theorem 2.7

We break the proof of Main Theorem (Theorem 2.7) into two parts, stated below. Notice that the reverse direction — that structured codes are LTCs — holds even for non-smooth (structured) codes.

Lemma 2.11 (Structured codes are LTCs)**.**

- *If $C \subseteq \mathbb{F}^n$ is $(q, \epsilon, \rho, \rho')$-structured then C is a $(q, \epsilon^2\rho, \rho')$-LTC. Furthermore, if C is t-smooth and structured then it is t-smooth as an LTC.*
- *If $C \subseteq \mathbb{F}^n$ is $(q, \epsilon, \rho, \rho')$-strongly structured then C is a $(q, \epsilon^2\rho\rho')$-strong LTC. Furthermore, if C is t-smooth and strongly structured then it is t-smooth as a strong LTC.*

We notice that Lemma 2.11 does not assume that the code C has a linear distance but only that it is (strongly) structured. The proof of Lemma 2.11 appears in Section 4. The following lemma states the forward direction of Theorem 2.7.

Lemma 2.12 (Smooth LTCs are structured)**.** *Let $C \subseteq \mathbb{F}^n$ be a (q, ϵ, ρ, t)-smooth LTC such that $\rho \leq \delta(C)/3$. Then C is $(q, \frac{\epsilon^2}{120tq^2(\log|\mathbb{F}|)}, \frac{\epsilon^3}{7\cdot 120tq(\log|\mathbb{F}|)}, 2\rho)$-structured and $(120tq(\log|\mathbb{F}|)/\epsilon^2)$-smooth.*

The proof of Lemma 2.12 is postponed to Section 5.

Proof of Main Theorem 2.7. Let $\mathcal{F} = \{C_{n_i}\}_{i=1,2,\ldots}$ be a family of linear codes. We argue that $\mathcal{F}$ is smooth q-LTC if and only if $\mathcal{F}$ is smooth q-structured.

For the first direction, assume that $C \subseteq \mathbb{F}^n$ is a (q, ϵ, ρ, t)-smooth LTC such that $\rho \leq \delta(C)/3$. Then Lemma 2.12 implies that C is $(q, \frac{\epsilon^2}{120tq^2(\log|\mathbb{F}|)}, \frac{\epsilon^3}{7\cdot 120tq(\log|\mathbb{F}|)}, 2\rho)$-structured and $(120tq(\log|\mathbb{F}|)/\epsilon^2)$-smooth.

For the second direction, assume that $C \subseteq \mathbb{F}^n$ is a $(q, \epsilon, \rho, \rho')$-structured and t-smooth. Then by Lemma 2.11 it follows that C is a $(q, \epsilon^2\rho, \rho', t)$-smooth LTC. □

3 Testing of small cyclic-width linear properties

We want to define a measure, called cwidth, with regards to the constraints of the code. Imagine that the coordinates $[n]$ are associated with elements of $\mathbb{Z}_n$, the cyclic group of size n. Notice that there can be different permutations over $[n]$ and for each such permutation the code remains the same, up to a different enumeration of the coordinates. For permutation $\sigma : [n] \to [n]$ and a vector $t \in \mathbb{F}^n$ we denote by $\sigma(t)$ the vector permuted by σ. Similarly, for subset $T \subseteq \mathbb{F}^n$ we define $\sigma(T) = \{\sigma(t) \mid t \in T\}$. Note that $\sigma(T) \subseteq \mathbb{F}^n$.

Intuitively, the cyclic-width of a constraint is the width of a "window" required to view all its support. More formally, the cyclic-width of a constraint $u \in \mathbb{F}_2^n$ is a minimal integer $s \in [n]$ such that for some $i \in [n]$ it holds that $\mathrm{supp}(u) \subseteq \{(i+1) \mod n, (i+2) \mod n, \ldots, (i+s)\} \mod n$, i.e.,

$$\mathrm{cwidth}(u) = \min\{s \in [n] | \exists i \in [n] : \mathrm{supp}(u) \subseteq (\{(i+1), (i+2), \ldots, (i+s)\} \mod n)\}.$$

Notice that if $\mathrm{supp}(u) = \{1, n\}$ for $u \in \mathbb{F}_2^n$ then $\mathrm{cwidth}(u) = 2$.

For $S \subseteq \mathbb{F}^n$ we let $\mathrm{cwidth}(S) = \min_{\sigma:[n]\to[n]} \left(\max_{u\in\sigma(S)} \{\mathrm{cwidth}(u)\} \right)$. In words, the cyclic-width of a subset is obtained by choosing the optimal arrangement of coordinates to receive the minimal

upper-bound on the cyclic-width of all vectors in S. We say that $\mathrm{cwidth}(S) = m$ with respect to σ if $m = \left(\max_{u \in \sigma(S)} \{\mathrm{cwidth}(u)\}\right)$.

It turns out that there is a relation between unique neighbor expanding Tanner graphs and the cyclic-width of constraints in the graph.

3.1 Expansion implies large cyclic-width

Assume $C \subseteq \mathbb{F}^n$ is a linear code and let $S \subseteq C^\perp$. For a subset $L_0 \subseteq [n]$ let $N_S^1(L_0) = \{u \in S \mid |\mathrm{supp}(u) \cap L_0| = 1\}$. We say that a code C has (ϵ, δ)-unique neighbor expansion if for every $L_0 \subseteq [n]$ if $|L_0| \leq \delta n$ then $|N_S^1(L_0)| \geq \epsilon |L_0|$. The code is called c-regular if for every $i \in [n]$ we have $|\{u \in S \mid i \in \mathrm{supp}(u)\}| \leq c$.

In the following claim we assume $C_1^\perp = \emptyset$, i.e., the code does not contain entries which are identically 0. This can be assumed since such entries can always be removed from the code.

Claim 3.1. *Let $\epsilon, \delta, c > 0$ be constants. Assume that a linear code C is c-regular and has (ϵ, δ)-unique neighbor expansion with respect to $S \subseteq C^\perp$ such that $C_1^\perp = \emptyset$. Then,*

$$\mathrm{cwidth}(S) \geq \frac{\epsilon \delta n}{4c} = \Omega(n).$$

Proof. Assume by contradiction that $\mathrm{cwidth}(S) < \frac{\epsilon \delta n}{4c}$. Then for every $u \in S$ we have $\mathrm{cwidth}(u) \leq \mathrm{cwidth}(S) < \frac{\epsilon \delta n}{4c}$. Let $L_0 = [\delta n/2]$. It holds that $|N_S^1(L_0)| \leq 2 \cdot \mathrm{cwidth}(S) \cdot c < \epsilon |L_0|$ with contradiction to the expansion property of C. In the above inequality we used two facts. The first one is that every unique neighbor of L_0 has cyclic-width at most $\mathrm{cwidth}(S)$ and thus only the indices of $\{1, 2, \ldots, \mathrm{cwidth}(S)\}$ and $\{|L_0| - \mathrm{cwidth}(S) + 1, |L_0| - \mathrm{cwidth}(S) + 2, \ldots, |L_0|\}$ can be contained in the support of any $u \in N_S^1(L_0)$. The second fact is that every index i can be contained in at most c constraints of $N_S^1(L_0)$. □

In the rest of the section we will be interested in a linear code C with $S \subseteq C^\perp$ such that $\mathrm{span}(S) = C^\perp$ and $\mathrm{cwidth}(S) = o(n)$.

3.2 A relation to the work of Ben-Sasson et al. [13]

Ben-Sasson et al. [13] showed that a random regular LDPC code $C \subseteq \mathbb{F}_2^n$ is not testable, and in particular, its testing requires $\Omega(n)$ queries. This random LDPC code C was picked by random selection of many constant weight dual codewords $S = \{u_{j_1}, u_{j_2}, \ldots\}$ and defining $C = (\mathrm{span}(S))^\perp$. Then it was examined that the fact that if small weight dual codewords in S are randomly chosen then they have two basic properties, which imply the impossibility of testing the code C. The first property was large unique neighbor expansion which holds for random Tanner graphs with high probability. This property, in particular, implies large cyclic-width of the code (Claim 3.1). The second property was that if many dual codewords in S are summed, then the resulting dual codeword will necessarily have large support size. This second property means that, relative to the number of dual codewords used in a summation, the number of cancelations is not too large.

In Definition 3.2 we require a similar property: every dual codeword can be spanned only by "adjacent" dual codewords of S. This property is similar to the above second property since it also requires assumes a small number of cancelations in the sum.

Definition 3.2 (d-adjacent vectors)**.** Let $u_1, u_2 \in \mathbb{F}_2^n$ be vectors. We say that u_1 is d-adjacent to u_2 if for every $i_1 \in u_1$ there exists $i_2 \in u_2$ such that $\min\{|i_1 - i_2| + 1, n - |i_1 - i_2| + 1\} \leq d$. Note that the "adjacency of vectors" is not a symmetric property, i.e., it might be that u_1 is d-adjacent to u_2 but u_2 is not d-adjacent to u_1.

We say that u_1 is d-adjacent to u_2 with respect to a permutation $\sigma : [n] \to [n]$ if $\sigma(u_1)$ is d-adjacent to $\sigma(u_2)$.

We say that $B \subseteq C^\perp$ is d-neighboring with respect to C and permutation σ if for $u \in C^\perp$ we have $u_1, \ldots, u_m \in B$ such that $\sum_{i=1}^{m} u_i = u$ and for all $i \in [m]$ it holds that u_i is d-adjacent to u with respect to σ.

Now we state Theorem 3.3.

Theorem 3.3 (Codes with small cyclic-width constraints are testable)**.** *Let $C \subseteq \mathbb{F}^n$ be a linear subspace. Assume that there exists a permutation $\sigma : [n] \to [n]$ and a subset $B \subseteq C^\perp$ such that $\mathrm{span}(B) = C^\perp$ and B is d-neighboring with respect to σ and $\mathrm{cwidth}(B) \leq m$ with respect to σ. Then C is $(4m, \frac{1}{6}, 1, \frac{1}{m \cdot (\lceil d/m \rceil)})$-strongly structured and by Lemma 2.11 is a $(4m, \frac{1}{36 \cdot \lceil d/m \rceil m})$-strong LTC.*

The proof of Theorem 3.3 appears in Section 7.

Example 3.4. Let $C \subseteq \mathbb{F}^{\log n}$ be any linear code. Assume that $(\log n) | n$. Let $C' \subseteq \mathbb{F}^n$ be a linear code such that for every $j \in [n/\log n]$ it holds that $C'|_{i,i+1,\ldots,\log n} = C$. Let $B = \bigcup_{j \in [n/\log n]} C'|_{i,i+1,\ldots,\log n}{}^\perp$ and let $\mathcal{D}$ be a distribution over S' that picks first a random $j \in [n/\log n]$ and then a random dual codeword in $C'|_{i,i+1,\ldots,\log n}{}^\perp$.

It can be verified that B is $(\log n)$-neighboring with respect to the identity permutation and $\mathrm{cwidth}(B) \leq \log n$ with respect to the identity permutation.

Notice that the distance of C is only $O(\log n)$ but let us stress that locally testable codes with very small distance (and even constant absolute Hamming distance) found their applications in [7].

Remark 3.5. Note that if C is a (q, ϵ)-strong LTC then C is in particular a $(\frac{q}{\epsilon}, \frac{1}{2})$-LTC since a tester (which has only one-sided error) for C can be sampled $\frac{1}{\epsilon}$ times and reject if at least one of its samples rejects. This implies query complexity $\frac{q}{\epsilon}$ and the soundness parameter is improved from ϵ to $\frac{1}{2}$.

Thus if C is a $(4m, \frac{1}{36\lceil d/m \rceil m})$-strong LTC then it is a $(36 \cdot 4 \cdot \lceil d/m \rceil m, \frac{1}{2})$-strong LTC.

In particular, an interesting range is when $d, m = 2^{o(\log n)}$ and then $\lceil d/m \rceil m = o(n)$, i.e., the query complexity of the tester of C is sublinear while the rejection probability is constant.

Remark 3.6. We were unable to resolve in this paper whether Theorem 3.3 holds even if "d-neighboring" is not required, i.e., whether the following conjecture holds.

Conjecture 3.7. *Let $B \subseteq C^\perp$ such that $\mathrm{span}(B) = C^\perp$ and $\mathrm{cwidth}(B) \leq m$. Then C is $(4m, \frac{1}{\mathrm{poly}(m)}, 1, \frac{1}{\mathrm{poly}(m)})$-strongly structured and thus locally testable.*

3.3 A relation to the work of Lachish et al. [33]

We recall Conjecture 1.2 restated next. Throughout this paper we consider only deterministic space complexity. Our model for measuring the space complexity of the algorithm is the standard Turing

Machine model, where there is a read only input tape, and a work tape where the machine can write. We only count the space used by the work tape. See [39] for the precise definitions.

Conjecture 1.2 (restated)**.** For any function $s(n)$ and a linear language $L \in \mathbf{DSPACE}(s(n))$ it holds that L is testable with $2^{O(s(n))}$ queries.

Now, every linear subspace $L_n \subseteq \mathbb{F}_2^n$ can be described in terms of linear constraints. Intuitively, without loss of generality, we can assume that the Turing Machine M that decides the language L checks linear constraints, i.e., given $x \in \mathbb{F}_2^n$ the machine M checks whether $x \perp (L_n)^\perp$. More accurately, for some subset $S_n \subseteq (L_n)^\perp$ such that $\mathrm{span}(S_n) = L_n$ the machine M checks whether $x \perp S_n$.

The question is which kind of linear constraints can be checked by deterministic Turing Machine with space complexity $s(n) = o(\log n)$. While we don't provide in this paper an answer on this question, we would like to present some evidence in support of it. Notice that if a memory tape is bounded to $s(n) = o(\log n)$ bits then the number of all possible contents on this tape is upper-bounded by $2^{s(n)} = o(n)$. In this way, this Turing Machine can "count" only up to $2^{s(n)} = o(n)$. In particular, even after the first $2^{s(n)} = o(n)$ moves of M some content of the memory tape will appear at least twice. Hence during the run of the machine M on a word we expect some particular memory-contents to reappear many times. This kind of argument has been used to prove "pumping" lemmas for regular and context free languages, and was a central observation used to show limitations of sublogarithmic computations [34, 2, 3]. Notably, it was proved [2, 3] that $\log n$ is a lower bound on the space complexity for the recognition of any deterministic non-regular context free language. I.e., there are no non-regular deterministic context free languages in $\mathbf{NSPACE}(o(\log n))$[4]. It is worth to stress that some non-regular deterministic context free languages belongs to $\mathbf{DSPACE}(O(\log n))$, e.g., the Dyck-languages [27, 40] and the parenthesis-languages [35, 36]. Hence there exists a natural distinction between $\mathbf{DSPACE}(O(\log n))$ and $\mathbf{DSPACE}(o(\log n))$.

Informally, we assume that the linear constraints that can be checked by Turing Machine with space complexity $s(n)$ either have cyclic-width $2^{s(n)}$ or should be periodic (like xor of all bits, or xor of all even bits). Now, such periodic constraints should be either in the span of small cyclic-width constraints, or quite different from any vector in the span of small cyclic-width constraints. We formalize this as follows.

Let $u, v \in \mathbb{F}^n$ be vectors and $m \in \mathcal{N}$ be a positive integer such that $m \leq n$. We say that u is m-different from v if $u|_{[m]} \neq v|_{[m]}$. For a subset of vectors $B_1, B_2 \subseteq \mathbb{F}^n$ we say that B_2 is m-different from B_1 if every $u \in \mathrm{span}(B_2) \setminus \{0^n\}$ is m-different from every $v \in \mathrm{span}(B_1)$. In particular, this means that for every $u \in \mathrm{span}(B_2) \setminus \{0^n\}$ we have $u|_{[m]} \neq 0^m$.

Theorem 3.8 (Codes with small cyclic-width and different constraints are testable)**.** *Assume $C \subseteq \mathbb{F}_2^n$ is linear and let $m = m(n), d = d(n)$ such that $0 < m, d = o(n)$. Assume that $B_1, B_2 \subseteq C^\perp$ such that B_1 is a d-neighboring subset, $\mathrm{cwidth}(B_1) \leq m$, and B_2 is m-different from B_1. Assume that $\mathrm{span}(B_1 \cup B_2) = C^\perp$.*

Then C is a $(4m, \frac{1}{6}, \frac{1}{20m\cdot(\lceil d/m \rceil)}, \frac{1}{10})$-structured and by Lemma 2.11 is a $(4m, \frac{1}{36\cdot 20m\cdot\lceil d/m \rceil}, \frac{1}{10})$-LTC.

The proof of Theorem 3.8 is postponed to Section 8. Similarly to Theorem 3.3 and Remark 3.6 it seems that the "d-neighboring" requirement is redundant, and the following statement should hold.

[4]This statement was first proved for $\mathbf{DSPACE}(o(\log n))$ and then was extended to $\mathbf{NSPACE}(o(\log n))$.

Conjecture 3.9. *Assume $C \subseteq \mathbb{F}_2^n$ is linear and let $m = m(n) > 0$. Assume that $B_1, B_2 \subseteq C^\perp$ such that $\mathrm{cwidth}(B_1) \leq m$, and B_2 is m-different from B_1. Assume that $\mathrm{span}(B_1 \cup B_2) = C^\perp$. Then C is $(4m, \frac{1}{\mathrm{poly}(m)}, \frac{1}{\mathrm{poly}(m)}, \Theta(1))$-structured and thus locally testable.*

If Conjecture 3.9 turns out to be true, then Conjecture 1.2 holds under the following conjecture.

Conjecture 3.10. *If $\{C_n\}_{n\in\mathcal{N}} \in$ **DSPACE**$(s(n))$, where $s(n) = o(\log n)$, then there exists $m(n) = 2^{O(s(n))}$ and for every $n \in \mathcal{N}$ there exist $B_1, B_2 \subseteq C^\perp$ such that $\mathrm{span}(B_1 \cup B_2) = C^\perp$, $\mathrm{cwidth}(B_1) \leq m$ and B_2 is m-different from B_1.*

Conjectures 3.9 and 3.10 now imply Conjecture 1.2.

Corollary 3.11. *Under Conjectures 3.9 and 3.10, Conjecture 1.2 holds.*

Proof. A linear language can be viewed as a family of linear codes $\{C_n\}_{n\in\mathcal{N}}$. If $\{C_n\}_{n\in\mathcal{N}} \in$ **DSPACE**$(s(n))$, where $s(n) = o(\log n)$, then Conjectures 3.9 and 3.10 imply that for every n the code C_n is $(4m, \frac{1}{\mathrm{poly}(m)}, \frac{1}{\mathrm{poly}(m)}, \Theta(1))$-structured for $m = 2^{O(s(n))}$ and by Lemma 2.11 is testable with $2^{O(s(n))}$ queries. □

4 Structured codes are LTCs — Proof of Lemma 2.11

To prove Lemma 2.11 we need the following definition.

Definition 4.1 (ϵ-bad set)**.** Let $C \subseteq \mathbb{F}^n$ be a linear code and $\mathcal{D}$ be a distribution supported on $S \subseteq C^\perp$. Let $i \in [n]$ such that $\mathcal{D}(N_S(i)) > 0$. For $w \in \mathbb{F}^n$ let $\mathrm{Bad}_i(w) = \{u \in N_S(i) \mid \langle u, w\rangle \neq 0\}$. We say i is an ϵ-bad entry of w if $\Pr_{u\sim\mathcal{D}}[u \in Bad_i(w) \mid i \in \mathrm{supp}(u)] \geq \epsilon$. Otherwise, i is called ϵ-good. Similarly, let $\mathrm{Bad}(w, \epsilon)$ denote the set of ϵ-bad entries of w.

Now we state Proposition 4.2 which will follows immediately from Definition 4.1.

Proposition 4.2. *Let $C \subseteq \mathbb{F}^n$ be a code and let $\mathcal{D}$ be its associated distribution supported on $S \subseteq C^\perp_{\leq q}$. Then if C is $(q, \epsilon, \rho, \rho')$-strongly structured and $w \in \mathbb{F}^n$, then*

$$\Pr_{u\sim\mathcal{D}}[\langle u, w\rangle \neq 0] \geq \epsilon^2 \cdot \frac{|\mathrm{Bad}(w, \epsilon)|}{n}.$$

Moreover, if C is $(q, \epsilon, \rho, \rho')$-structured and $w \in \mathbb{F}^n$ such that $|\mathrm{Bad}(w, \epsilon)| \geq \rho n$, then

$$\Pr_{u\sim\mathcal{D}}[\langle u, w\rangle \neq 0] \geq \epsilon^2 \cdot \frac{|\mathrm{Bad}(w, \epsilon)|}{n}.$$

Proof. By definition of $\mathrm{Bad}(w, \epsilon)$ (Definition 4.1) we have

$$\Pr_{u\sim\mathcal{D}}[\langle u, w\rangle \neq 0] \geq \Pr_{u\sim\mathcal{D}}[\langle u, w\rangle \neq 0 \mid \mathrm{supp}(u) \cap \mathrm{Bad}(w, \epsilon) \neq \emptyset] \cdot \Pr_{u\sim\mathcal{D}}[\mathrm{supp}(u) \cap \mathrm{Bad}(w, \epsilon) \neq \emptyset] \geq$$

$$\geq \epsilon \cdot \Pr_{u\sim\mathcal{D}}[\mathrm{supp}(u) \cap \mathrm{Bad}(w, \epsilon) \neq \emptyset].$$

By the sampling property of Definition 2.5 and the fact that C is (strongly) structured it follows that

$$\Pr_{u\sim\mathcal{D}}[\mathrm{supp}(u) \cap B_w \neq \emptyset] \geq \epsilon \cdot \frac{|\mathrm{Bad}(w, \epsilon)|}{n}.$$

We conclude that $\Pr_{u\sim\mathcal{D}}[\langle u, w\rangle \neq 0] \geq \epsilon^2 \cdot \frac{|\mathrm{Bad}(w, \epsilon)|}{n}$. □

We are ready to prove Lemma 2.11.

Proof of Lemma 2.11. Let $S \subseteq C^{\perp}_{\leq q}$ and $\mathcal{D}$ be as guaranteed by the definition of $(q, \epsilon, \rho, \rho')$-structured code (Definition 2.5) and let $w \in \mathbb{F}^n$. Set $J = \mathrm{Bad}(w, \epsilon)$. Note that the fact that J is a set of ϵ-bad entries implies that for all $i \in J$ there exists $u \in S$ such that $i \in \mathrm{supp}(u)$.

For the first bullet of the lemma assume that $\delta(w, C) \geq \rho'$. We prove that $\mathop{\mathbf{Pr}}_{u \sim \mathcal{D}}[\langle u, w \rangle \neq 0] \geq \rho\epsilon^2$.

We claim that $|J| \geq \rho n$. To see this, assume by contradiction that $|J| < \rho n$. We argue that for all $u \in S_{(-J)}$ we have $\langle u, w \rangle = 0$. To see this, let $S' = \{u' \in S \mid \langle u', w \rangle \neq 0\}$ and let $u \in S_{(-J)}$, i.e., $\mathrm{supp}(u) \cap J = \emptyset$. We have $\dfrac{\mathcal{D}(S' \cap N_S(i))}{\mathcal{D}(N_S(i))} < \epsilon$ for all $i \in \mathrm{supp}(u)$ since otherwise $\mathrm{supp}(u)$ contains an ϵ-bad entry of w and $\mathrm{supp}(u) \cap J \neq \emptyset$. By assumption C is ϵ-redundant and hence $u \in \mathrm{span}(S \setminus S')$ and $\langle u, w \rangle = 0$. Recall that $|J| < \rho n$ and C is ρ'-strictly characterized by $S_{(-J)}$. Hence by Claim A.2 we have $\delta(w, C) < \rho'$. Contradiction.

So $|J| \geq \rho n$. Thus Proposition 4.2 implies that

$$\mathop{\mathbf{Pr}}_{u \sim \mathcal{D}}[\langle u, w \rangle \neq 0] \geq \epsilon^2 \cdot \frac{|J|}{n} \geq \epsilon^2 \rho$$

and we are done.

For the second bullet of the lemma, assume now that C is $(q, \epsilon, \rho, \rho')$-strongly structured and set $J = \mathrm{Bad}(w, \epsilon)$ as above. If $|J| \geq \rho n$ then by Proposition 4.2 we have $\mathop{\mathbf{Pr}}_{u \sim \mathcal{D}}[\langle u, w \rangle \neq 0] \geq \epsilon^2 \rho$. Otherwise, $|J| < \rho n$ and C is $\left(\frac{|J|}{n} \cdot \frac{1}{\rho'}\right)$-characterized by $S_{(-J)}$. Then, by Claim A.2 we have $\delta(w, C) \leq \left(\frac{|J|}{n} \cdot \frac{1}{\rho'}\right) < \frac{\rho}{\rho'}$ and $\mathop{\mathbf{Pr}}_{u \sim \mathcal{D}}[\langle u, w \rangle \neq 0] \geq \epsilon^2 \dfrac{|J|}{n} \geq \epsilon^2 \cdot \rho' \cdot \delta(w, C)$.

Hence $\mathop{\mathbf{Pr}}_{u \sim \mathcal{D}}[\langle u, w \rangle \neq 0] \geq \epsilon^2 \cdot \min\{\rho, \rho'\} \cdot \delta(w, C) \geq \epsilon^2 \cdot \rho \cdot \rho' \cdot \delta(w, C)$.

Finally, notice that if C is t-smooth as a structured code, then the tester for C is also t-smooth. □

5 Smooth LTCs are structured — Proof of Lemma 2.12

We notice that if S is a multiset then $N_S(i)$ is a multiset and $|N_S(i)|$ counts the repetitions.

Overview of the proof. First, we state and prove Proposition 5.1. Informally, it says that without loss of generality a tester for a smooth LTC $C \subseteq \mathbb{F}^n$ is a uniform distribution over a multiset of size $\Theta(n)$ such that every dual codeword of this multiset has a small number of neighbor dual codewords in this multiset.

The proof of Proposition 5.1 looks as follows. First, a tester for the linear code $C \subseteq \mathbb{F}^n$ is sampled $O(n)$ times to obtain a multiset of small weight dual codewords S. It turns out that with high probability a uniform distribution over S is a tester for C. Let $S' \subseteq S$ be a subset of dual codewords that contain all $u \in S$ such that $|N_u(S)|$ is large. Then we prove that with high probability the set S' is small. Finally, letting $S^* = (S \setminus S')$ we argue that a uniform distribution over S^* is a tester for C and for every $u \in S^*$ it holds that $|N_{S^*}(u)|$ is small.

Then we prove Lemma 2.12, and in particular we prove that C is structured with respect to S^* and the uniform distribution over S^*.

Proposition 5.1. *Let $C \subseteq \mathbb{F}^n$ be a (q, ϵ, ρ, t)-smooth LTC. Then there exists a multiset $S^* \subseteq C^{\perp}_{\leq q}$ such that for all $u \in S^*$ we have $|N_{S^*}(u)| \leq 120tq(\log|\mathbb{F}|)/\epsilon^2$ and C has a $(q, \epsilon/6, \rho)$-tester, which is uniformly distributed over S^*. Moreover,*

$$(20n(\log|\mathbb{F}|)/\epsilon)(1-\epsilon/3) \leq |S^*| \leq 20n(\log|\mathbb{F}|)/\epsilon.$$

Proof. Let T be the assumed (q, ϵ, ρ, t)-smooth tester for C. Pick the multiset $S = \left\{u_1, \ldots, u_{\frac{20n(\log|\mathbb{F}|)}{\epsilon}}\right\}$, where each u_j obtained by taking a sample from T. We know that $|S| = \frac{20n(\log\mathbb{F})}{\epsilon}$ and for every $u_j \in S$ we have $|u_j| \leq q$. We argue that the uniform distribution over the multiset S will be a $(q, \epsilon/2, \rho)$-tester for C with probability at least $1 - \frac{1}{|\mathbb{F}|^{0.1n}}$.

Fix any $w \in \mathbb{F}^n$ such that $\delta(w, C) \geq \rho$. We know that $\mathbf{Pr}_{u\in S}[\langle u, w\rangle \neq 0] \geq \epsilon$ and hence $\mathbf{E}[|\{u \in S \mid \langle u, w\rangle \neq 0\}|] \geq \epsilon|S|$. Chernoff bound (Claim 5.2) implies that

$$\mathbf{Pr}\Big[|\{u \in S \mid \langle u, w\rangle \neq 0\}| \leq \frac{\epsilon}{2}|S|\Big] \leq \frac{1}{2^{0.25\epsilon|S|/3}} \leq \frac{1}{|\mathbb{F}|^{1.1n}}.$$

Take the union bound over all $w \in \mathbb{F}^n$ such that $\delta(w, C) \geq \rho$ to get that with probability at most $\frac{|\mathbb{F}|^n}{|\mathbb{F}|^{1.1n}}$ there exists a word $w \in \mathbb{F}^n$, $\delta(w, C) \geq \rho$ such that $\mathbf{Pr}_{u\in S}[\langle u, w\rangle \neq 0] < \epsilon/2$. We conclude that with probability at least $1 - \frac{1}{|\mathbb{F}|^{0.1n}}$ we have that for every $w \in \mathbb{F}^n$, $\delta(w, C) \geq \rho$ it holds that $\mathbf{Pr}_{u\in_U S}[\langle u, w\rangle \neq 0] \geq \epsilon/2$, i.e., the uniform distribution over the multiset S is a $(q, \epsilon/2, \rho)$-tester for C.

Since C is a (q, ϵ, ρ, t)-smooth LTC it follows that for every $i \in [n]$ we have

$$\mathbf{E}[|\{u \in S \mid i \in \mathrm{supp}(u)\}|] \leq |S| \cdot \frac{t}{n} = 20t/\epsilon.$$

For any $u \in S$ let X_u be a random variable defined by $X_u = |N_S(u)|$. We have

$$\mathbf{E}[X_u] \leq |S| \cdot q \cdot \frac{t}{n} = 20tq(\log|\mathbb{F}|)/\epsilon,$$

where the expectation is taken over the samplings of S. Markov's inequality implies that

$$\mathbf{Pr}\left[X_u > 120tq(\log|\mathbb{F}|)/\epsilon^2\right] < \epsilon/6.$$

Let X_S be a random variable defined by

$$X_S = \left|\left\{u \in S \mid X_u > 120tq(\log|\mathbb{F}|)/\epsilon^2\right\}\right|.$$

Hence $\mathbf{E}[X_S] \leq (\epsilon/6)|S|$. Markov's inequality implies that

$$\mathbf{Pr}[X_S \geq (\epsilon/3)|S|] \leq \frac{1}{2}.$$

Let $S' = \left\{u \in S \mid |N_S(u)| \geq 120tq(\log|\mathbb{F}|)/\epsilon^2\right\}$, note that $S' \subseteq S$. We have $|S'| \leq \epsilon|S|/3$ with probability at least $\frac{1}{2}$.

Hence with probability at least $1 - \frac{1}{2} - \frac{1}{|\mathbb{F}|^{0.1n}} > 0$ there exist S and S' as required. Fix them and let $S^* = S \setminus S'$. Note that

$$\frac{20n(\log|\mathbb{F}|)}{\epsilon} \cdot (1-\epsilon/3) \leq |S^*| \leq \frac{20n(\log|\mathbb{F}|)}{\epsilon}.$$

We know that for all $u \in S^*$ we have $|N_{S^*}(u)| \leq 120tq(\log|\mathbb{F}|)/\epsilon^2$. Moreover, for all $w \in \mathbb{F}^n$ such that $\delta(w, C) \geq \rho$ we have

$$\mathop{\mathbf{Pr}}_{u\in S^*}[\langle u, w\rangle \neq 0] \geq \mathop{\mathbf{Pr}}_{u\in S}[\langle u, w\rangle \neq 0] - \mathop{\mathbf{Pr}}_{u\in S}\left[u \in S'\right] = \mathop{\mathbf{Pr}}_{u\in S}[\langle u, w\rangle \neq 0] - \frac{|S'|}{|S|} \geq \epsilon/2 - \epsilon/3 = \epsilon/6.$$

We conclude that the uniform distribution over the multiset S^* is a $(q, \epsilon/6, \rho)$-tester for C. □

For the sake of completeness we state the particular version of Chernoff's inequality that we use.

Claim 5.2 (Chernoff Bound)**.** *If $X = \sum_{i=1}^{m} X_i$ is a sum of independent $\{0,1\}$-valued random variables, where $\mathbf{Pr}[X_i = 1] = \gamma$, then*

$$\mathbf{Pr}[X < (1-\sigma)\gamma m] \leq \exp(-\frac{\sigma^2\gamma m}{3}) \quad \textit{and} \quad \mathbf{Pr}[X > (1+\sigma)\gamma m] \leq \exp(-\frac{\sigma^2\gamma m}{3}).$$

Now we prove Lemma 2.12.

Proof of Lemma 2.12. Proposition 5.1 guarantees that there exists a multiset $S \subseteq C^{\perp}_{\leq q}$ such that

$$\text{for every } \ u \in S \ \text{ we have } \ |N_S(u)| \leq \frac{120tq(\log|\mathbb{F}|)}{\epsilon^2},$$

$$\frac{20n(\log|\mathbb{F}|)}{\epsilon} \cdot (1-\epsilon/3) \leq |S| \leq \frac{20n(\log|\mathbb{F}|)}{\epsilon},$$

and for all $w \in \mathbb{F}^n, \delta(w, C) \geq \rho$ we have $\mathop{\mathbf{Pr}}_{u\in_U S}[\langle u, w\rangle \neq 0] \geq \epsilon/6$. In particular, this implies that for all $i \in [n]$ we have $|N_S(i)| \leq 120tq(\log|\mathbb{F}|)/\epsilon^2$.

Hence if $\langle u, w\rangle \neq 0$ for $u \in S$ then every $i \in \mathrm{supp}(u)$ is a $\dfrac{\epsilon^2}{120tq(\log|\mathbb{F}|)}$-bad entry of w because $u \in N_S(u)$ and $|N_S(u)| \leq \dfrac{120tq(\log|\mathbb{F}|)}{\epsilon^2}$. Thus C is $\left(\dfrac{\epsilon^2}{120tq(\log|\mathbb{F}|)}\right)$-redundant with respect to S and $\mathcal{D}$.

Let $J \subseteq [n]$ such that $|J| \leq \dfrac{\epsilon^3}{7 \cdot 120tq(\log|\mathbb{F}|)}n$. Then the number of words $u \in S$ such that $\mathrm{supp}(u) \cap J \neq \emptyset$ is at most

$$\left(120tq(\log|\mathbb{F}|)/\epsilon^2\right) \cdot |J| \leq (\epsilon/7) \cdot n \leq (\epsilon/7) \cdot |S|.$$

Then $S_{(-J)}$ is still a $(q, \epsilon/6 - \epsilon/7, \rho)$-tester for C because removing all words of S whose support intersect J can reduce the rejection probability of the tester at most by $\dfrac{(\epsilon/7) \cdot |S|}{|S|} = \epsilon/7$. That means if $w \perp S_{(-J)}$ then $\delta(w, C) < \rho$, i.e., Corollary A.3 implies that C is (2ρ)-characterized by $S_{(-J)}$.

Furthermore, for every $I \subseteq [n]$ such that $\forall i \in I, \exists u \in S : i \in \mathrm{supp}(u)$ we have

$$\mathop{\mathbf{Pr}}_{u\sim\mathcal{D}}[\mathrm{supp}(u) \cap I \neq \emptyset] \geq \frac{\epsilon^2}{120tq(\log|\mathbb{F}|)} \cdot \frac{1}{q} \cdot \frac{|I|}{n}.$$

We conclude that C is $(q, \frac{\epsilon^2}{120tq^2(\log|\mathbb{F}|)}, \frac{\epsilon^3}{7\cdot 120tq(\log|\mathbb{F}|)}, 2\rho)$-structured. Moreover, for every $j \in [n]$ we have

$$\mathop{\mathbf{Pr}}_{u\in_U S}[j \in \mathrm{supp}(u)] \leq \frac{120tq(\log|\mathbb{F}|)/\epsilon^2}{n}.$$

□

6 Hadamard codes are strongly structured — Proof of Theorem 2.9

In this section we show that the family of Hadamard codes (cf. Definition 2.8) is strongly structured.

We thus set $C \subseteq \mathbb{F}_2^n$ to be the Hadamard code and $S = C_3^\perp$. It is known that $\mathrm{span}(S) = C^\perp$ and $\Delta(C^\perp) = 3$, i.e., for all $u_1 \neq u_2 \in S$ it holds that $|\mathrm{supp}(u_1) \cap \mathrm{supp}(u_2)| \leq 1$. As a tester for the Hadamard code we take the uniform distribution $\mathcal{U}$ over S, i.e., given an input word $w \in \mathbb{F}_2^n$ the tester picks $u \in_{\mathcal{U}} S$ and accepts if and only if $\langle u, w\rangle = 0$.

We know that $|S| = \frac{\binom{n}{2}}{3} = \frac{n(n-1)}{6}$ since every two different coordinates $i_1, i_2 \in [n]$ determines uniquely $u \in S$ such that $i_1, i_2 \in \mathrm{supp}(u)$. Let us state and prove the following auxiliary claim.

Claim 6.1. *For $I \subseteq [n]$ and $j \in I$ let $A_{(j,I)} = \{u \in C_3^\perp \mid \mathrm{supp}(u) \cap I = \{j\}\}$ and $B_{(j,I)} = \{u \in C_3^\perp \mid j \in \mathrm{supp}(u), |\mathrm{supp}(u) \cap I| \geq 2\}$. Then,*

- $|A_{(j,\{j\})}| = \frac{n-1}{2}$,
- $|B_{(j,I)}| \leq |I| - 1$, *and*
- $|A_{(j,I)}| = |A_{(j,\{j\})}| - |B_{(j,I)}| \geq \frac{n-2|I|+1}{2}$.

Proof. Since $\Delta(C^\perp) = 3$ there are no $u_1 \neq u_2 \in C_3^\perp$ such that $\mathrm{supp}(u_1) \cap \mathrm{supp}(u_2) \geq 2$. For every $i_1 \neq i_2 \in [n]$ there exists unique $u \in C_3^\perp$ such that $i_1, i_2 \in \mathrm{supp}(u)$. Thus given $j \in [n]$ we have $n-1$ options for $j' \in [n] \setminus \{j\}$ such that for some $u \in C_3^\perp$ we have $j, j' \in \mathrm{supp}(u)$. Since every $u \in C_3^\perp$ with $\mathrm{supp}(u) = \{j, j', j''\}$ was counted twice, once from j' and once from j'' we have $|A_{(j,\{j\})}| = |\{u \in C_3^\perp \mid j \in \mathrm{supp}(u)\}| = \frac{n-1}{2}$.

We have $|B_{(j,I)}| \leq |I| - 1$ because for every $j' \in I$ such that $j' \neq j$ we have only one $u \in C_3^\perp$ with $j, j' \in \mathrm{supp}(u)$. Hence $|A_{(j,I)}| = |A_{(j,\{j\})}| - |B_{(j,I)}| \geq \frac{n-2|I|+1}{2}$. □

Now we state Lemmas 6.2 and 6.3. Then we prove Theorem 2.9.

Lemma 6.2. *It holds that C is $\frac{1}{3}$-redundant with respect to S and $\mathcal{U}$.*

Lemma 6.3. *It holds that C is $(\frac{1}{6}, 1)$-strongly stable with respect to S.*

The proofs of Lemmas 6.2 and 6.3 appear in Sections 6.1 and 6.2. We are ready to prove Theorem 2.9.

Proof of Theorem 2.9. Recall that $S = C_3^\perp$. Lemma 6.2 says that C is $\frac{1}{3}$-redundant with respect to S and $\mathcal{U}$. Lemma 6.3 implies that C is $(\frac{1}{6}, 1)$-strongly stable with respect to S. It holds that for all $I \subseteq [n]$ we have

$$\mathop{\mathbf{Pr}}_{u \sim \mathcal{U}}[\mathrm{supp}(u) \cap I \neq \emptyset] \geq \frac{|I|}{n} \geq \frac{1}{3} \cdot \frac{|I|}{n}.$$

Moreover, we claim that Hadamard is 3-smooth because for all $j \in [n]$ we have

$$\mathop{\mathbf{Pr}}_{u \in_{\mathcal{U}} S}[j \in \mathrm{supp}(u)] = \frac{|A_{(j,\{j\})}|}{|S|} = \frac{(n-1)/2}{n(n-1)/6} = \frac{3}{n}.$$

We conclude that Hadamard is $(3, \frac{1}{3}, \frac{1}{6}, 1)$-strongly structured 3-smooth code. □

6.1 Proof of Lemma 6.2

Recall that $S = C_3^\perp$ and let us prove Lemma 6.2.

Proof of Lemma 6.2. Let $u \in S$. Let $S' \subseteq S$ such that for all $j \in \mathrm{supp}(u)$ we have $|N_S(j) \cap S'| \leq \frac{1}{3}|N_S(j)|$. It is sufficient to prove that $u \in \mathrm{span}(S \setminus S')$.

Recall that

$$N_S(u) = \left\{u' \in C_3^\perp \mid \mathrm{supp}(u) \cap \mathrm{supp}(u') \neq \emptyset\right\} = \left\{u' \in C_3^\perp \mid |\mathrm{supp}(u) \cap \mathrm{supp}(u')| = 1\right\} \cup \{u\}.$$

Assume without loss of generality that $\mathrm{supp}(u) = \{n-2, n-1, n\}$. For each $a \in \{0,1,2\}$ let

$$A_a = N_S(n-a) \setminus \{u\} = \{v \in S \mid (n-a) \in \mathrm{supp}(v), v \neq u\}.$$

Claim 6.1 implies that $|A_0| = |A_1| = |A_2| = \frac{n-1}{2} - |\{u\}| = \frac{n-3}{2}$. Notice that A_0, A_1, A_2 are disjoint and $A_0 \cup A_1 \cup A_2 \cup \{u\} = N_S(u)$, i.e., $A_0, A_1, A_2, \{u\}$ is a partition of $N_S(u)$.

Let $m = \frac{n-3}{2}$. We prove that the elements of A_0, A_1 and A_2 can be ordered to satisfy the following condition.

Condition 6.4.

- $A_0 = \left\{u_0^{(0)}, u_1^{(0)}, \ldots, u_m^{(0)}\right\}$,
- $A_1 = \left\{u_0^{(1)}, u_1^{(1)}, \ldots, u_m^{(1)}\right\}$,
- $A_2 = \left\{u_0^{(2)}, u_1^{(2)}, \ldots, u_m^{(2)}\right\}$, and
- for all $i \in [m]$ we have $u = u_0^{(i)} + u_1^{(i)} + u_2^{(i)}$.

We argue that if there exists such an ordering then we are done. If $u \notin S'$ then we are done because $u \in S \setminus S'$ and hence $u \in \mathrm{span}(S \setminus S')$. Otherwise $u \in S'$ and moreover, $u \in N_S(j)$ for all $j \in \mathrm{supp}(u)$. We conclude that $|A_a \cap S'| < \frac{1}{3}|A_a|$ for all $a \in \{0,1,2\}$. Hence $|(A_0 \cup A_1 \cup A_2) \cap S'| < |A_0|$ and so, there exists $i \in [m]$ such that $u_0^{(i)}, u_1^{(i)}, u_2^{(i)} \notin S'$, which implies that $u \in \mathrm{span}(S \setminus S')$.

Now we prove that the elements of A_0, A_1 and A_2 can be ordered as in Condition 6.4. Consider bipartite undirected graph $G = (L = A_0, R = A_1, E)$ where $(u_0, u_1) \in E$ iff $u_0 \in A_0$, $u_1 \in A_1$ and $(u + u_0 + u_1) \in A_2$). For $V \subseteq A_0 \cup A_1$ let $N(V)$ be a set of neighbors of V in G. For $v \in A_0 \cup A_1$ let $\deg(v)$ be a degree of v in G, i.e., $\deg(v) = |N(\{v\})|$.

It is sufficient to show that for all $v \in A_0 \cup A_1$ we have $\deg(v) = 2$, because in this case for all $L_0 \subseteq L$ it holds that $|N(L_0)| \geq |L_0|$. This is true since the assumption that $|N(L_0)| < |L_0|$ for some $L_0 \subseteq L$ implies that for some $v_0 \in L_0$ and $v_1 \in N(L_0)$ it holds that $\deg(v_0) < \deg(v_1)$. Contradiction. Now we use Hall's Theorem (Theorem 6.5) to conclude that G has perfect matching.

We show that every node $v \in A_0 \cup A_1$ has degree 2. Because of the symmetry it is sufficient to show that $\deg(v) = 2$ for all $v \in A_0$. So, let $v \in A_0$ and assume that $\mathrm{supp}(v) = \{n, i_1, i_2\}$, where $i_1 \neq i_2 \in [n-1]$. Clearly, v has only one neighbor $v' \in A_1$ such that $\{i_1, n-1\} \subset \mathrm{supp}(v')$ and one neighbor $v'' \in A_1$ such that $\{i_2, n-1\} \subset \mathrm{supp}(v'')$. Note that if $v^* \in A_1$ such that $\mathrm{supp}(v^*) \cap \mathrm{supp}(v) = \emptyset$ then $u + v + v^* \notin A_2$ and hence $(v, v^*) \notin E$.

Consider the perfect matching that we have in G and for every $u_i^{(0)} \in A_0$ let $u_i^{(1)} \in A_1$ be the "matched" node. Reorder the elements of A_0 and A_1 such that $A_0 = \left\{u_0^{(0)}, u_1^{(0)}, ..., u_m^{(0)}\right\}$, $A_1 = \left\{u_0^{(1)}, u_1^{(1)}, ..., u_m^{(1)}\right\}$ and for all $i \in [m]$ we have that $u_i^{(0)}$ is matched to $u_i^{(1)}$. Note that for all $i_1 \neq i_2 \in [m]$ we have that $u + u_{(i_1)}^0 + u_{(i_1)}^1, u + u_{(i_2)}^0 + u_{(i_2)}^1 \in A_2$ and $u + u_{(i_1)}^0 + u_{(i_1)}^1 \neq u + u_{(i_2)}^0 + u_{(i_2)}^1$. Hence we can reorder the elements of A_2 such that $A_2 = \left\{u_0^{(2)}, u_1^{(2)}, ..., u_m^{(2)}\right\}$ and for all $i \in [m]$ we have that $u = u_i^{(0)} + u_i^{(1)} + u_i^{(2)}$. The lemma follows. □

For the sake of completeness we provide the Hall's theorem.

Theorem 6.5 (Hall's theorem)**.** *Let $G = (L, R, E)$ be a bipartite undirected graph. Then perfect matching exists if and only if $\forall L_0 \subseteq L$ we have $|N(L_0)| \geq |L_0|$, where $N(V) = \{u \mid \exists v \in V : (u, v) \in E\}$ is set of neighbors of V.*

6.2 Proof of Lemma 6.3

Proof of Lemma 6.3. We prove that C is $(\frac{1}{6}, 1)$-strongly stable with respect to S. Let $I \subset [n]$ such that $|I| \leq \frac{1}{6} \cdot n$. We should prove that C is $\frac{|I|}{n}$-characterized by $S_{(-I)}$. By Claim A.2, it is sufficient to show that for all $w \in \mathbb{F}_2^n$ such that $w \perp S_{(-I)}$ it holds that $w|_{[n]\setminus I} \in C|_{[n]\setminus I}$.

Let $w \in \mathbb{F}_2^n$ such that $w \perp S_{(-I)}$. We prove that $w|_{[n]\setminus I} \in C|_{[n]\setminus I}$, by showing the existence of $c \in C$ such that $\mathrm{supp}(w - c) \subseteq I$ and conclude that $w|_{[n]\setminus I} = c|_{[n]\setminus I} \in C|_{[n]\setminus I}$.

Note that if $c \perp S$ then $c \in (\mathrm{span}(S))^\perp = (C^\perp)^\perp = C$. Let $I = \{i_1, \ldots, i_m\}$ and $w_0 = w$. Note that $m \leq n/6$. We prove that it is possible to fix all bits of $(w_0)|_I$ to get w_m such that $w_m \perp S$.

We prove this by induction on $i_j \in I$. For the basis, recall that $w_0 \perp S_{(-I)}$. Assume that we have fixed the bits $i_1, \ldots, i_{j-1}$ of w, obtaining w_{j-1} such that $\mathrm{supp}(w - w_{j-1}) \subseteq \{i_1, \ldots, i_{j-1}\}$ and $w_{j-1} \perp S_{(-\{i_j, \ldots, i_m\})}$.

We prove that it is possible to fix the bit i_j of w_{j-1}, obtaining w_j, such that $w_j \perp S_{(-\{j+1, \ldots, i_m\})}$. Let $I' = \{i_j, \ldots, i_m\}$ and note that $|I'| \leq |I| \leq n/6$.

Let $J = \left\{u \in C_3^\perp \mid \mathrm{supp}(u) \cap I' = \{i_j\}\right\}$. Claim 6.1 implies that $|J| \geq \dfrac{n - 2|I'| + 1}{2}$. Similarly, let $J_0 = \{u \in J \mid \langle u, w_{j-1}\rangle = 0\}$ and let $J_1 = \{u \in J \mid \langle u, w_{j-1}\rangle \neq 0\}$. We know that $J_0 \cap J_1 = \emptyset$ and $J_0 \cup J_1 = J$. So, either $|J_0| \geq |J|/2$ or $|J| \geq |J_0|/2$. Assume without loss of generality that $|J_1| \geq |J|/2$, i.e.,

$$|J_1| \geq \frac{n - 2|I'| + 1}{4} \geq \frac{n - 2|I| + 1}{4} > n/6.$$

If we have $J = J_1$ then flip the jth bit of w_{j-1}, obtaining w_j such that $\langle w_j, u\rangle = 0$ for all $u \in S_{(-\{j+1, \ldots, i_m\})}$.

Hence assume that there exists $u_0 \in J_0$ such that $\langle u_0, w_{j-1}\rangle = 0$. We show that there exist $u \in J_1$ and $u', u'' \in S_{(-I')}$ such that $u_0 + u + u' + u'' = 0$. This results in a contradiction, because $\langle u_0, w\rangle = \langle u', w\rangle = \langle u'', w\rangle = 0$ and $u_0 + u + u' + u'' = 0$ but $\langle u, w\rangle \neq 0$.

Recall that we have $u_0 \in J_0$ and let $\mathrm{supp}(u_0) = \{i_j, j', j''\}$ such that $j', j'' \notin I'$ and $\langle u_0, w\rangle = 0$. We consider all vectors $u \in J_1$. Note that $|J_1| > n/6$. For $u \in J_1$, $\mathrm{supp}(u) = \{i_j, j_1, j_2\}$ we have $j_1, j_2 \notin I'$. Every such $u \in J_1$ with $\mathrm{supp}(u) = \{i_j, j_1, j_2\}$ and u_0 uniquely defines vector $u' \in S$ such that $j', j_1 \in \mathrm{supp}(u')$. Notice that the number of vectors $u' \in S$ such that $j' \in \mathrm{supp}(u')$ and $\mathrm{supp}(u') \cap I' \neq \emptyset$ is at most $|I'| \leq n/6$. We conclude that there exists $u \in J_1$ with $\mathrm{supp}(u) = \{i_j, j_1, j_2\}$ and $u' \in S$ such that $j', j_1 \in \mathrm{supp}(u')$ and $\mathrm{supp}(u') \cap I' = \emptyset$. This is true since j' is fixed and we have $|J_1| > |I'|$.

Let $u'' = u_0 + u + u'$ and thus $u_0 + u + u' + u'' = 0$. We argue that $u'' \in S_{(-I')}$. To see this note that $\mathrm{supp}(u_0) = \{i_j, j', j''\}$, $\mathrm{supp}(u) = \{i_j, j_1, j_2\}$, $\mathrm{supp}(u') \supset \{j', j_1\}$, $u' \in S_{(-I')}$ and $j_1, j_2, j', j'' \notin I'$. The Lemma follows. □

7 Proof of Theorem 3.3

Let us start with an auxiliary definition of blocks. Without loss of generality assume that $m|n$. For $i \in [n/m]$ let us define the ith block by

$$\mathrm{block_i} = \{\mathrm{m} \cdot (\mathrm{i} - 1) + 1, \ldots, \mathrm{m} \cdot (\mathrm{i} - 1) + \mathrm{m}\}.$$

Note that these blocks provide a partition of coordinates in $[n]$. Notice that if $\mathrm{cwidth}(B) \leq m$ then for every $u \in B$ we have $\mathrm{supp}(u) \subseteq \mathrm{block_i} \cup \mathrm{block_{i+1}}$ or $\mathrm{supp}(u) \subseteq \mathrm{block_{(n/m)}} \cup \mathrm{block_1}$, i.e., the support is included in some two subsequent blocks. This explains the intuition behind the definition of block and why its size was chosen to be m.

Remark 7.1. In case n is not divisible by m, the size of the last block will be between m and $2m$. In particular, one can see that the above property holds as well as other statements in the rest of the paper.

Definition 7.2 (Adjacent Blocks)**.** Given $I \subseteq [n]$ we let

$$\mathrm{blocks(I)} = \{\mathrm{i} \in [\mathrm{n}] \mid \exists \mathrm{j} : \mathrm{i} \in \mathrm{block_j}, \mathrm{I} \cap \mathrm{block_j} \neq \emptyset\}.$$

For $u \in C^{\perp}$, with some abuse of notation we let $\mathrm{blocks(u)} = \mathrm{blocks(supp(u))}$.

For $j \in [n/m]$, if $j < n/m$ we let $\mathrm{NextBlock(block_j)} = \mathrm{block_{(j+1)}}$ and otherwise ($j = n/m$) we let $\mathrm{NextBlock(block_j)} = \mathrm{block_1}$. Similarly, we define a previous block: $\mathrm{PrevBlock(block_j)} = \mathrm{block_i}$ iff $\mathrm{NextBlock(block_i)} = \mathrm{block_j}$.

We also define NextBlocks and PrevBlocks for a subset $I \subseteq [n]$ by

$$\mathrm{NextBlocks(I)} = \{\mathrm{i} \in [\mathrm{n}] \mid \exists \mathrm{j_1}, \mathrm{j_2} : \mathrm{i} \in \mathrm{block_{j_2}}, \mathrm{I} \cap \mathrm{block_{j_1}} \neq 0, \mathrm{block_{j_2}} = \mathrm{NextBlock(block_{j_1})}\}$$

and

$$\mathrm{PrevBlocks(I)} = \{\mathrm{i} \in [\mathrm{n}] \mid \exists \mathrm{j_1}, \mathrm{j_2} : \mathrm{i} \in \mathrm{block_{j_2}}, \mathrm{I} \cap \mathrm{block_{j_1}} \neq 0, \mathrm{block_{j_2}} = \mathrm{PrevBlock(block_{j_1})}\}.$$

Let $\mathrm{Adjacent(I)} = \mathrm{blocks(I)} \cup \mathrm{NextBlocks(I)} \cup \mathrm{PrevBlocks(I)}$. We define $\mathrm{Adjacent^d}$ by induction on d. We let $\mathrm{Adjacent^1(I)} = \mathrm{Adjacent(I)}$ and $\mathrm{Adjacent^d(I)} = \mathrm{Adjacent(Adjacent^{d-1}(I))}$.

Now we define a subset S of small dual codewords. Let

$$S = \left\{ u \in C^{\perp}_{\leq 2m} \mid \exists j : \mathrm{supp}(u) \subseteq \mathrm{block_j} \cup \mathrm{NextBlock}(\mathrm{block_j}) \right\}.$$

Note that $B \subseteq S$ and for all $u \in S$ we have $|u| \leq 2m$ and $\mathrm{cwidth}(u) \leq 2m$.

As pointed out in Remark 7.1, if n is not divisible by m, then for all $u \in S$ we have $|u| \leq 4m$ and $\mathrm{cwidth}(u) \leq 4m$. This is the reason why we state Theorem 3.3 with "$4m$" and not with $2m$. However, for the convenience, in the rest of the proof we will think that all blocks have size m.

We will define a distribution $\mathcal{D}$ over S such that for every $u \in S$ we have $\mathcal{D}(u) > 0$. We define the distribution $\mathcal{D}$ over S by defining the sampling of $u \in S$.

- pick random $j \in [n/(2m)]$
- pick random $u \in (C|_{(\mathrm{block_j} \cup \mathrm{NextBlock}(\mathrm{block_j}))})^{\perp}$.

Note that for all $J \subseteq [n], J \neq \emptyset$ it holds that $C|_J{}^{\perp} \neq \emptyset$ since $0^{|J|} \in C|_J{}^{\perp}$.

The following claim is immediate.

Claim 7.3. *Assume that $B \subseteq C^{\perp}$ is d-neighboring with respect to C. Then for every $u \in C^{\perp}$ we have $u_1, \ldots, u_m \in B$ such that $\sum_{i=1}^{m} u_i = u$ and for all $i \in [m]$ it holds that* $\mathrm{supp}(u_i) \subseteq \mathrm{Adjacent}^{\lceil \mathrm{d/m} \rceil}(\mathrm{supp(u)})$.

Proof. The proof follows since if u_i is d-adjacent to u then $\mathrm{supp}(u_i) \subseteq \mathrm{Adjacent}^{\lceil \mathrm{d/m} \rceil}(\mathrm{supp(u)})$. □

Let us state a couple of lemmas. Then we will prove Theorem 3.3.

Lemma 7.4. *It holds that C is $\frac{1}{6}$-redundant with respect to S and $\mathcal{D}$.*

Lemma 7.5. *It holds that C is $(1, \frac{1}{m \cdot (\lceil d/m \rceil)})$-strongly stable with respect to S and $\mathcal{D}$.*

The proofs of the Lemmas are postponed to Section 7.1. Now we prove Theorem 3.3.

Proof of Theorem 3.3. Lemma 7.4 implies that C is $\frac{1}{6}$-redundant with respect to S and $\mathcal{D}$. Lemma 7.5 implies that C is $\left(1, \frac{1}{m \cdot (\lceil d/m \rceil)}\right)$-strongly stable.

Moreover, for all $I \subseteq [n]$ such that $\forall i \in I, \exists u \in S : i \in \mathrm{supp}(u)$ we have

$$\Pr_{u \sim \mathcal{D}}[\mathrm{supp}(u) \cap I \neq \emptyset] \geq \frac{1}{6} \cdot \frac{|I|}{n}.$$

Hence C is $(4m, \frac{1}{6}, 1, \frac{1}{m \cdot (\lceil d/m \rceil)})$-strongly structured with respect to S and $\mathcal{D}$. □

7.1 Proofs of Lemmas 7.4 and 7.5

Claim 7.6. *Let $w \in \mathbb{F}_2^n$ and $u \in S$ such that $\langle u, w \rangle \neq 0$. Assume that $J \subseteq [n]$ such that* $\mathrm{supp}(u) \subseteq J$. *Let $i \in \mathrm{supp}(u)$. Then*

$$\left| \left\{ u' \in C^{\perp} \mid \mathrm{supp}(u') \subseteq J, i \in \mathrm{supp}(u'), \langle u', w \rangle \neq 0 \right\} \right| \geq \frac{1}{2} \cdot \left| \left\{ u' \in C^{\perp} \mid \mathrm{supp}(u') \subseteq J, i \in \mathrm{supp}(u') \right\} \right|.$$

Proof. Let us define the following subsets

$$C_{\notin,0} = \left\{u' \in C^{\perp} \mid \text{supp}(u') \subseteq J, i \notin \text{supp}(u'), \langle u', w\rangle = 0\right\}$$

$$C_{\in,0} = \left\{u' \in C^{\perp} \mid \text{supp}(u') \subseteq J, i \in \text{supp}(u'), \langle u', w\rangle = 0\right\}$$

$$C_{\notin,\neq 0} = \left\{u' \in C^{\perp} \mid \text{supp}(u') \subseteq J, i \notin \text{supp}(u'), \langle u', w\rangle \neq 0\right\}$$

and

$$C_{\in,\neq 0} = \left\{u' \in C^{\perp} \mid \text{supp}(u') \subseteq J, i \notin \text{supp}(u'), \langle u', w\rangle \neq 0\right\}.$$

In the rest of the Claim we prove that $\left|C_{\notin,0}\right| = |C_{\in,\neq 0}|$ and then we prove that $\left|C_{\notin,0}\right| = |C_{\in,0}|$. We conclude that $|C_{\in,\neq 0}| = |C_{\in,0}|$. This implies the Claim since

$$|C_{\in,\neq 0}| = \left|\left\{u' \in S \mid \text{supp}(u') \subseteq J, i \in \text{supp}(u'), \langle u', w\rangle \neq 0\right\}\right| \geq$$

$$\frac{1}{2} \cdot (|C_{\in,\neq}| + |C_{\in,0}|) = \frac{1}{2} \cdot \left|\left\{u' \in C^{\perp} \mid \text{supp}(u') \subseteq J, i \in \text{supp}(u')\right\}\right|.$$

Now, let us prove that $|C_{\notin,0}| = |C_{\in,\neq 0}|$. We know that $P|_J$ is a linear subspace, $0^n \in C_{\notin,0}$ and $u \in C_{\in,\neq 0}$. We know that for every $u' \in C_{\in,\neq 0}$ we have $u + u' \in C_{\notin,0}$, and for every $v' \in C_{\notin,0}$ we have $u + v' \in C_{\notin,0}$. We conclude that $|C_{\notin,0}| = |C_{\in,\neq 0}|$.

Now, we prove that $|C_{\notin,0}| = |C_{\in,0}|$. If $C_{\in,0} = \emptyset$ we are done. Otherwise, assume that $v \in C_{\in,0}$ and recall that $0^n \in C_{\notin,0}$. We know that for all $u' \in C_{\notin,0}$ we have $u' + v \in C_{\in,0}$ and for all $v' \in C_{\in,0}$ we have $v + v' \in C_{\notin,0}$. Thus $|C_{\notin,0}| = |C_{\in,0}|$. □

We are ready to prove Lemma 7.4.

Proof of Lemma 7.4. Let $w \in \mathbb{F}_2^n$ and $u \in S$. Let $i \in \text{supp}(u)$. By Claim A.4, it is sufficient to prove that if $\langle u, w\rangle \neq 0$ then

$$\mathop{\mathbf{Pr}}_{u' \sim \mathcal{D}}\left[\langle u', w\rangle \neq 0 \mid i \in \text{supp}(u')\right] \geq \frac{1}{6}.$$

So, we assume that $\langle u, w\rangle \neq 0$. By definition of $\mathcal{D}$ we know that $\text{supp}(u) \subseteq \text{block}_j \cup \text{NextBlock}(\text{block}_j)$ for some j. Then $i \in \text{block}_j \cup \text{NextBlock}(\text{block}_j)$.

If vector u' is sampled according to $\mathcal{D}$, then given that $i \in \text{supp}(u')$ with probability at least $\frac{1}{3}$ we have $\text{supp}(u') \subseteq \text{block}_j \cup \text{NextBlock}(\text{block}_j)$ since there are only three possible (not necessary disjoint) events:

- $i \in \text{PrevBlock}(\text{block}_j) \cup \text{block}_j$,
- $i \in \text{block}_j \cup \text{NextBlock}(\text{block}_j)$,
- $i \in \text{NextBlock}(\text{block}_j) \cup \text{NextBlock}(\text{NextBlock}(\text{block}_j))$;

and each one of pairs of blocks is chosen with the same probability.

Using the fact that $\text{supp}(u') \subseteq \text{block}_j \cup \text{NextBlock}(\text{block}_j)$ and $i \in \text{block}_j \cup \text{NextBlock}(\text{block}_j)$, by Claim 7.6 we have

$$\mathop{\mathbf{Pr}}_{u' \sim \mathcal{D}}\left[\langle u', w\rangle \neq 0 \mid i \in \text{supp}(u'), \text{supp}(u') \subseteq \text{block}_j \cup \text{NextBlock}(\text{block}_j)\right] \geq \frac{1}{2}.$$

Letting $Blocks = \text{block}_j \cup \text{NextBlock}(\text{block}_j)$ we conclude that

$$\Pr_{u' \sim \mathcal{D}}\left[\langle u', w\rangle \neq 0 \mid i \in \text{supp}(u')\right] =$$

$$\Pr_{u' \sim \mathcal{D}}\left[\langle u', w\rangle \neq 0 \mid i \in \text{supp}(u'), \text{supp}(u') \subseteq Blocks\right] \cdot \Pr_{u' \sim \mathcal{D}}\left[\text{supp}(u') \subseteq Blocks \mid i \in \text{supp}(u')\right] \geq$$

$$\geq \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{6}.$$

□

Now we prove Lemma 7.5.

Proof of Lemma 7.5. Let $I \subseteq [n]$ be any subset (intuitively, think that I is a set of bad bits). Let $J = \text{Adjacent}^{(\lceil d/m \rceil)}(\text{blocks(I)})$ and note that

$$|J| \leq m \cdot (\lceil d/m \rceil) \cdot |I|,$$

because in the worst case every coordinate $i \in I$ belongs to the separate block (of size m) and hence $|\text{blocks(I)}| \leq |\text{I}| \cdot \text{m}$. Thus

$$|\text{Adjacent}^{(\lceil \text{d/m} \rceil)}(\text{blocks(I)})| \leq (\lceil \text{d/m} \rceil) \cdot |\text{blocks(I)}| \leq |\text{I}| \cdot \text{m} \cdot (\lceil \text{d/m} \rceil).$$

Let $u \in C^{\perp}$ such that $\text{supp}(u) \cap J = \emptyset$. We prove that $u \in \text{span}\, S_{(-I)}$. By Definition 3.2 (recall that $B \subseteq S$ is d-neighboring) and Claim 7.3, there exists $u_1, \ldots, u_k \in S$ such that $u_1 + \ldots + u_k = u$ and $\text{supp}(u_i) \cap I = \emptyset$ for all $i \in [k]$. Thus $(C^{\perp})_{(-J)} \subseteq \text{span}\, S_{(-I)}$.

We showed that for every $I \subseteq [n]$ there exists $J \subseteq [n]$, $|J| \leq m \cdot (\lceil d/m \rceil) \cdot |I|$ such that $(C^{\perp})_{(-J)} \subseteq \text{span}\, S_{(-I)}$. We conclude that C is $(1, \frac{1}{m \cdot (\lceil d/m \rceil)})$-strongly stable with respect to S and $\mathcal{D}$. □

8 Proof of Theorem 3.8

Let $C_1 = \text{span}(B_1)^{\perp}$.

First of all, we consider the code C_1 with regards to B_1. The set of small weight dual codewords will be defined by $S \subseteq C^{\perp}_{\leq 4m}$ and the distribution $\mathcal{D}$ over S, are defined exactly as in the proof of Theorem 3.3.

It follows that C_1 is $\frac{1}{6}$-redundant with regards to S and $\mathcal{D}$ (the proof is the same to the proof of Lemma 7.4). Similarly to the proof of Theorem 7.4, for all $I \subseteq [n]$ such that $\forall i \in I, \exists u \in S : i \in \text{supp}(u)$ we have

$$\Pr_{u \sim \mathcal{D}}[\text{supp}(u) \cap I \neq \emptyset] \geq \frac{1}{6} \cdot \frac{|I|}{n}.$$

Let $J = [m]$. Now, we argue that $C|_{[n]\setminus J} = (C_1)|_{[n]\setminus J}$.

By definition, $((C_1)|_{[n]\setminus J})^{\perp} \subseteq (C|_{[n]\setminus J})^{\perp}$. It remains to show that $(C|_{[n]\setminus J})^{\perp} \subseteq ((C_1)|_{[n]\setminus J})^{\perp}$. We know that for every $u \in C^{\perp} \setminus \text{span}(B_1)$ it holds that $u = u_1 + u_2$ for some $u_1 \in \text{span}(B_1)$ and $u_2 \in \text{span}(B_2) \setminus \{0^n\}$. We have $u_1, u_2 \neq 0^n$. By definition, u_2 is m-different from u_1 and that means $\text{supp}(u) \cap J \neq \emptyset$.

We conclude that for every $u' \in (C|_{[n]\setminus J})^{\perp}$ there exists $u \in {C_1}^{\perp}$ such that $u|_{[n]\setminus J} = u'$. Hence $(C|_{[n]\setminus J})^{\perp} \subseteq ((C_1)|_{[n]\setminus J})^{\perp}$.

Now we recall the proof of Lemma 7.5 to prove that C is $(\frac{1}{20m\cdot(\lceil d/m\rceil)}, \frac{1}{10})$-stable with regards to S. Let $I \subseteq [n]$ be any subset (intuitively, think that I is a set of bad bits). Assume that $|I| < \frac{1}{20m\cdot(\lceil d/m\rceil)} \cdot n$.

Let $J = [m] \cup \left(\text{Adjacent}^{(\lceil d/m \rceil)}(\text{blocks(I)})\right)$ (similarly to the proof of Lemma 7.5, Section 7) and note that

$$|J| \leq m + m \cdot (\lceil d/m \rceil) \cdot |I| \leq 2m \cdot (\lceil d/m \rceil) \cdot |I| < \frac{1}{10} \cdot n,$$

see the proof of Lemma 7.5 for the explanation.

By the proof of Lemma 7.5 and the fact that $C|_{[n]\setminus[m]} = (C_1)|_{[n]\setminus[m]}$, it follows that $(C^{\perp})_{(-J)} \subseteq \operatorname{span} S_{(-I)}$.

It follows that C is $(\frac{1}{20m\cdot(\lceil d/m\rceil)}, \frac{1}{10})$-stable with respect to S.

Acknowledgments

We thank Eyal Kushilevitz for pointers to the literature.

References

[1] Noga Alon, Michael Krivelevich, Ilan Newman, and Mario Szegedy. Regular languages are testable with a constant number of queries. *SIAM J. Comput*, 30(6):1842–1862, 2000.

[2] Helmut Alt. Lower bounds on space complexity for contextfree recognition. *Acta Informatica*, 12:33–61, 1979.

[3] Helmut Alt, Viliam Geffert, and Kurt Mehlhorn. A lower bound for the nondeterministic space complexity of context-free recognition. *Inf. Process. Lett*, 42(1):25–27, 1992.

[4] Sanjeev Arora. *Probabilistic checking of proofs and hardness of approximation problems.* PhD thesis, Princeton University, 1994.

[5] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof Verification and the Hardness of Approximation Problems. *Journal of the ACM*, 45(3):501–555, May 1998.

[6] Sanjeev Arora and Shmuel Safra. Probabilistic Checking of Proofs: A New Characterization of NP. *Journal of the ACM*, 45(1):70–122, January 1998.

[7] Boaz Barak, Parikshit Gopalan, Johan Håstad, Raghu Meka, Prasad Raghavendra, and David Steurer. Making the long code shorter, with applications to the unique games conjecture. *Electronic Colloquium on Computational Complexity (ECCC)*, 18:142, 2011.

[8] Mihir Bellare, Don Coppersmith, Johan Håstad, Marcos A. Kiwi, and Madhu Sudan. Linearity testing in characteristic two. *IEEE Transactions on Information Theory*, 42(6):1781–1795, 1996.

[9] Eli Ben-Sasson. Limitation on the rate of families of locally testable codes. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:123, 2010.

[10] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil P. Vadhan. Robust PCPs of Proximity, Shorter PCPs, and Applications to Coding. *SIAM Journal on Computing*, 36(4):889–974, 2006.

[11] Eli Ben-Sasson, Elena Grigorescu, Ghid Maatouk, Amir Shpilka, and Madhu Sudan. On sums of locally testable affine invariant properties. In *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX-RANDOM)*, volume 6845 of *Lecture Notes in Computer Science*, pages 400–411. Springer, 2011.

[12] Eli Ben-Sasson, Venkatesan Guruswami, Tali Kaufman, Madhu Sudan, and Michael Viderman. Locally Testable Codes Require Redundant Testers. *SIAM J. Comput*, 39(7):3230–3247, 2010.

[13] Eli Ben-Sasson, Prahladh Harsha, and Sofya Raskhodnikova. Some 3CNF Properties Are Hard to Test. *SIAM Journal on Computing*, 35(1):1–21, 2005.

[14] Eli Ben-Sasson, Ghid Maatouk, Amir Shpilka, and Madhu Sudan. Symmetric LDPC codes are not necessarily locally testable. In *IEEE Conference on Computational Complexity*, pages 55–65. IEEE Computer Society, 2011.

[15] Eli Ben-Sasson, Noga Ron-Zewi, and Madhu Sudan. Sparse affine-invariant linear codes are locally testable. *Electronic Colloquium on Computational Complexity (ECCC)*, 19:49, 2012.

[16] Eli Ben-Sasson and Madhu Sudan. Robust locally testable codes and products of codes. *Random Struct. Algorithms*, 28(4):387–402, 2006.

[17] Eli Ben-Sasson and Michael Viderman. Composition of Semi-LTCs by Two-Wise Tensor Products. In *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX-RANDOM)*, volume 5687 of *Lecture Notes in Computer Science*, pages 378–391. Springer, 2009.

[18] Eli Ben-Sasson and Michael Viderman. Tensor Products of Weakly Smooth Codes are Robust. *Theory of Computing*, 5(1):239–255, 2009.

[19] Eli Ben-Sasson and Michael Viderman. Low rate is insufficient for local testability. In *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX-RANDOM)*, volume 6302 of *Lecture Notes in Computer Science*, pages 420–433. Springer, 2010.

[20] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-Testing/Correcting with Applications to Numerical Problems. *J. Comput. Syst. Sci*, 47(3):549–595, 1993.

[21] Don Coppersmith and Atri Rudra. On the Robust Testability of Product of Codes. *Electronic Colloquium on Computational Complexity (ECCC)*, (104), 2005.

[22] Irit Dinur and Omer Reingold. Assignment Testers: Towards a Combinatorial Proof of the PCP Theorem. *SIAM Journal on Computing*, 36(4):975–1024, 2006.

[23] Irit Dinur, Madhu Sudan, and Avi Wigderson. Robust Local Testability of Tensor Products of LDPC Codes. In *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX-RANDOM)*, volume 4110 of *Lecture Notes in Computer Science*, pages 304–315. Springer, 2006.

[24] Katalin Friedl and Madhu Sudan. Some improvements to total degree tests. In *Proceedings of the 3rd Israel Symposium on Theory of Computing and Systems (ISTCS)*, pages 190–198, 1995.

[25] Oded Goldreich and Madhu Sudan. Locally testable codes and PCPs of almost-linear length. *Journal of the ACM*, 53(4):558–655, July 2006.

[26] J. E. Hopcroft and J. D. Ullman. Some results on tape-bounded turing machines. *Journal of the ACM*, 16(1):168–177, January 1969.

[27] Günter Hotz and Jan Messerschmidt. Dyck-sprachen sind in Bandkomplexität $\log n$ analysierbar. In *Universität des Saarlandes*, 1974.

[28] Tali Kaufman, Simon Litsyn, and Ning Xie. Breaking the epsilon-soundness bound of the linearity test over GF(2). *SIAM J. Comput*, 39(5):1988–2003, 2010.

[29] Tali Kaufman and Shachar Lovett. New extension of the weil bound for character sums with applications to coding. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 788–796. IEEE, 2011.

[30] Tali Kaufman and Madhu Sudan. Sparse Random Linear Codes are Locally Decodable and Testable. In *FOCS*, pages 590–600. IEEE Computer Society, 2007.

[31] Tali Kaufman and Madhu Sudan. Algebraic property testing: the role of invariance. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC), Victoria, British Columbia, Canada, May 17-20, 2008*, pages 403–412. ACM, 2008.

[32] Swastik Kopparty and Shubhangi Saraf. Tolerant Linearity Testing and Locally Testable Codes. In *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX-RANDOM)*, volume 5687 of *Lecture Notes in Computer Science*, pages 601–614. Springer, 2009.

[33] Oded Lachish, Ilan Newman, and Asaf Shapira. Space complexity vs. query complexity. *Computational Complexity*, 17(1):70–93, 2008.

[34] P. M. Lewis, R. E. Stearns, and J. Hartmanis. Memory bounds for recognition of context-free and context-sensitive languages. In *Proceedings of the 6th Annual Symposium on Switching Circuit Theory and Logical Design (SWCT 1965)*, pages 191–202, Washington, DC, USA, 1965. IEEE Computer Society.

[35] Nancy Lynch. Log space recognition and translation of parenthesis languages. *J. Assoc. Comput. Mach*, 24:583–590, 1977.

[36] Kurt Mehlhorn. Bracket-Languages are Recognizable in Logarithmic Space. *Inf. Process. Lett.*, pages 168–170, 1976.

[37] Or Meir. Combinatorial Construction of Locally Testable Codes. *SIAM J. Comput*, 39(2):491–544, 2009.

[38] Ilan Newman. Testing membership in languages that have small width branching programs. *SIAM J. Comput*, 31(5):1557–1570, 2002.

[39] Christos H. Papadimitriou. *Computational Complexity*. Addison-Wesley, New York, 1994.

[40] Robert W. Ritchie and Frederick N. Springsteel. Language Recognition by Marking Automata. *Information and Control*, pages 313–330, 1972.

[41] Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM J. Comput*, 25(2):252–271, 1996.

[42] Daniel Spielman. *Computationally efficient error-correcting codes and holographic proofs*. PhD thesis, Massachusetts Institute of Technology, 1996.

[43] Robert Michael Tanner. A recursive approach to low complexity codes. *IEEE Transactions on Information Theory*, 27(5):533–547, 1981.

[44] Paul Valiant. The Tensor Product of Two Codes Is Not Necessarily Robustly Testable. In *Proceedings of the Approximation, Randomization, and Combinatorial Optimization (APPROX-RANDOM)*, volume 3624 of *Lecture Notes in Computer Science*, pages 472–481. Springer, 2005.

A Auxiliary Claims

The proof of the next claim was implicit in [12, Claim 18].

Claim A.1. *Let $\rho > 0$. Let $V \subseteq \mathbb{F}^n$ and such that for all $v \in V$ we have $|v| < \rho n$. Assume that $|\bigcup_{v \in V} \mathrm{supp}(v)| \geq 2\rho$. Then there exists $w' \in \mathrm{span}(V)$ such that $\rho n \leq |w'| < 2\rho n$.*

Proof. Let $w_1, \ldots, w_s$ be an arbitrary ordering of the elements of V. Let $\mu(\ell)$ be the maximal weight of an element in $\mathrm{span}(w_1, \ldots, w_\ell)$. We have $\mu(1) = |w_1| < \rho n$ and $\mu(s) \geq \rho n$ because the expected weight of a word in $\mathrm{span}(V)$ is (exactly) $\frac{|\mathbb{F}|-1}{|\mathbb{F}|} \cdot \left|\bigcup_{w \in V}(\mathrm{supp}(w))\right| \geq \frac{1}{2} \cdot 2\rho n$. To see that the expected weight of a word $w_{exp} \in \mathrm{span}(V)$ is as claimed note that w_{exp} is picked by a random linear combination of vectors in V, where each vector $v \in V$ ia taken independently with probability $1 - \frac{1}{|\mathbb{F}|}$. Hence if $i \in \bigcup_{w \in V}(\mathrm{supp}(w))$ then $i \in \mathrm{supp}(w_{exp})$ with probability $\frac{|\mathbb{F}|-1}{|\mathbb{F}|} \geq \frac{1}{2}$. Thus there exist elements $\alpha_i \in \mathbb{F}$ such that $\sum_{i=1}^{s} \alpha_i \cdot w_i = w_{exp}$, where $|w_{exp}| \geq \rho n$, i.e., $\mu(s) \geq \rho n$.

Finally, we have $\mu(\ell+1) < \mu(\ell) + \rho n$ since for every $i \in [s]$ we have $|w_i| < \rho n$. We conclude there must exist ℓ for which $\rho n \leq \mu(\ell) < 2\rho n$. Let w' be a word of maximal weight in $\mathrm{span}(w_1, \ldots, w_\ell)$. We have that $\rho n \leq |w'| < 2\rho n$. □

Claim A.2. *It holds that $C \subseteq \mathbb{F}^n$ is ρ-characterized (ρ-strictly characterized) by S if and only if there exists $J' \subseteq [n]$, $|J'| \leq \rho n$ ($|J'| < \rho n$) and for all $w \perp S$ it holds that $w|_{[n]\setminus J'} \in C|_{[n]\setminus J'}$.*

Proof. For the first direction, if $C \subseteq \mathbb{F}^n$ is ρ-characterized (ρ-strictly characterized) by S then, by definition, there exists $J' \subseteq [n]$, $|J'| \leq \rho n$ ($|J'| < \rho n$) such that $|J'| \leq \rho n$ ($|J'| < \rho n$) and $(C^\perp)_{(-J')} \subseteq \mathrm{span}(S)$. This means that if $w \perp S$ then $w \perp (C^\perp)_{(-J')}$ and hence $w|_{[n]\setminus J'} \in C|_{[n]\setminus J'}$.

For the second direction, assume that there exists $J' \subseteq [n]$, $|J'| \leq \rho n$ ($|J'| < \rho n$) and for all $w \in \mathbb{F}^n$ such that $w \perp S$ it holds that $w|_{[n]\setminus J'} \in C|_{[n]\setminus J'}$. We argue that $(C^\perp)_{(-J')} \subseteq \mathrm{span}(S)$, since otherwise there exists $w \in \mathbb{F}^n$ such that $w \perp S$ and $w|_{[n]\setminus J'} \notin C|_{[n]\setminus J'}$. Hence C is ρ-characterized (ρ-strictly characterized) by S. □

Let $w \in \mathbb{F}^n$. We know that $\delta(w, C) \leq \rho$ if and only if there exists $J' \subseteq [n]$ such that $|J'| \leq \rho n$ and $w|_{[n]\setminus J'} \in (C)|_{[n]\setminus J'}$.

Corollary A.3. *Let $C \subseteq \mathbb{F}^n$ be a linear code and $S \subseteq C^\perp$. Let $\rho \leq \delta(C)/3$.*

- *If C is ρ-characterized (ρ-strictly characterized) by S then for all $w \in \mathbb{F}^n$ such that $w \perp S$ we have $\delta(w, C) \leq \rho$ ($\delta(w, C) < \rho$).*
- *If for all $w \in \mathbb{F}^n$ such that $w \perp S$ we have $\delta(w, C) < \rho$ then C is 2ρ-strictly characterized by S.*

Proof. The first bullet is straightforward. If C is ρ-characterized (ρ-strictly characterized) by S then, by Claim A.2 there exists $J' \subseteq [n]$, $|J'| \leq \rho n$ ($|J'| < \rho n$) and for all $w \perp S$ it holds that $w|_{[n]\setminus J'} \in (C)|_{[n]\setminus J'}$, i.e., $\delta(w, C) \leq \rho$ ($\delta(w, C) < \rho$).

For the second bullet, assume that for all $w \in \mathbb{F}^n$ such that $w \perp S$ we have $\delta(w, C) < \rho$. We argue that there exists a subset $J' \subseteq [n]$, $|J'| < 2\rho n$ such that for all $w \perp S$ we have $w|_{[n]\setminus J'} \in (C)|_{[n]\setminus J'}$. We say that w is a coset leader if w has minimal weight in $w + C = \{w + c \mid c \in C\}$. Notice that if $w \in \mathbb{F}^n$ is a coset leader then $\delta(w, C) = \delta(w, 0^n) = \frac{|w|}{n}$.

Let $J' = \{i \mid i \in \text{supp}(w) \text{ such that } w \in \mathbb{F}^n \text{ is a coset leader and } w \perp S\}$. By definition of J' we know that for all $w \perp S$ we have $w|_{[n]\setminus J'} \in C|_{[n]\setminus J'}$. To see this note that if $w' \in w + C$ is a coset leader then $w \perp S$ iff $w' \perp S$ and if $w \perp S$ then $w|_{[n]\setminus \text{supp}(w')} \in C|_{[n]\setminus \text{supp}(w')}$.

Let $V_{J'} = \{w \in \mathbb{F}^n \mid w \perp S, \text{w is a coset leader}\}$ and note that $\bigcup_{w \in V_{J'}} \text{supp}(w) = J'$. Note also that for all $w \in V_{J'}$ we have $|w| = n \cdot \delta(w, 0) \leq \rho n$, because w is a coset leader. It is sufficient to show that $|J'| < 2\rho n$. Assume the contradiction, i.e., $|J'| \geq 2\rho n$. Claim A.1 implies that there exists $w' \in \text{span}(V_{J'})$ such that $\rho n \leq |w'| \leq 2\rho n$. Recall that $\rho \leq \delta(C)/3$. Hence, $\delta(w', 0^n) \geq \rho$ and for all $c \in C \setminus \{0^n\}$ it holds that $\delta(w', c) \geq (|c| - |w'|)/n \geq \delta(C) - 2\delta(C)/3 = \delta(C)/3$. We conclude that $\delta(w', C) \geq \rho$. But $w' \perp S$, contradiction. □

Claim A.4. *Let $C \subseteq \mathbb{F}^n$, $S \subseteq C^\perp$ and $\mathcal{D}$ be a distribution over S. Then, C is ϵ-redundant with respect to S and $\mathcal{D}$ if and only if the following condition holds. For all $u \in S$ and $w \in \mathbb{F}^n$ if $\langle u, w \rangle \neq 0$ then there exists $i \in \text{supp}(u)$ such that i is ϵ-bad with respect to S and $\mathcal{D}$.*

Proof. For the first direction assume that C is ϵ-redundant with respect to S and $\mathcal{D}$. Let $u \in S$ and $w \in \mathbb{F}^n$ such that $\langle u, w \rangle \neq 0$. Let $S' = \{u' \in S \mid \langle u', w \rangle \neq 0\}$. Assume by contradiction that every $i \in \text{supp}(u)$ is good. Then for every $i \in \text{supp}(u)$ we have $\mathcal{D}(N_S(i) \cap S') < \epsilon \mathcal{D}(N_S(i))$ and hence $u \in \text{span}(S \setminus S')$ with contradiction to the fact that for all $u' \in S \setminus S'$ we have $\langle u', w \rangle = 0$ but $\langle u, w \rangle \neq 0$.

For the second direction, assume that for all $u \in S$ and $w \in \mathbb{F}^n$ if $\langle u, w \rangle \neq 0$ then there exists $i \in \text{supp}(u)$ such that i is ϵ-bad. Assume $S' \subseteq S$ such that for every $i \in \text{supp}(u)$ we have $\mathcal{D}(N_S(i) \cap S') < \epsilon \mathcal{D}(N_S(i))$. We argue that $u \in \text{span}(S \setminus S')$ since otherwise there exists $w \in \mathbb{F}^n$ such that $w \perp (S \setminus S')$, however $\langle u', w \rangle \neq 0$ for some $u' \in S'$. In this case, $\text{supp}(u)$ has no ϵ-bad index. Contradiction. □

ECCC ISSN 1433-8092
http://eccc.hpi-web.de