Testing Sumsets Is Hard

Xi Chen
Department of Computer Science, Columbia University, New York, NY, USA

Shivam Nadimpalli
Department of Mathematics, MIT, Cambridge, MA, USA

Tim Randolph
Department of Computer Science, Harvey Mudd College, Claremont, CA, USA

Rocco A. Servedio
Department of Computer Science, Columbia University, New York, NY, USA

Or Zamir
Blavatnik School of Computer Science, Tel Aviv University, Israel

Abstract

A subset S of the Boolean hypercube $\mathbb{F}_2^n$ is a *sumset* if $S = \{a + b : a, b \in A\}$ for some $A \subseteq \mathbb{F}_2^n$. Sumsets are central objects of study in additive combinatorics, where they play a role in several of the field's most important results. We prove a lower bound of $\Omega(2^{n/2})$ for the number of queries needed to test whether a Boolean function $f : \mathbb{F}_2^n \to \{0, 1\}$ is the indicator function of a sumset, ruling out an efficient testing algorithm for sumsets.

Our lower bound for testing sumsets follows from sharp bounds on the related problem of *shift testing*, which may be of independent interest. We also give a near-optimal $2^{n/2} \cdot \mathrm{poly}(n)$-query algorithm for a smoothed analysis formulation of the sumset *refutation* problem. Finally, we include a simple proof that the number of different sumsets in $\mathbb{F}_2^n$ is $2^{(1\pm o(1))2^{n-1}}$.

2012 ACM Subject Classification Theory of computation → Streaming, sublinear and near linear time algorithms

Keywords and phrases Sumsets, additive combinatorics, property testing, Boolean functions

Digital Object Identifier 10.4230/LIPIcs.ESA.2025.14

Related Version *Full Version*: `https://arxiv.org/abs/2401.07242`

Funding *Xi Chen*: Supported in part by NSF awards CCF-2106429 and CCF-2107187.
Shivam Nadimpalli: Supported in part by NSF grants CCF-2106429, CCF-2211238, CCF-1763970, and CCF-210718.
Rocco A. Servedio: Supported in part by NSF awards CCF-2211238 and CCF-2106429.
Or Zamir: Supported in part by the Israel Science Foundation, Grant No. 1593/24, and in part by the Blavatnik Family Foundation.

1 Introduction

In recent years, theoretical computer science has increasingly been influenced by ideas and techniques from *additive combinatorics*, a field sitting at the intersection of combinatorics, number theory, Fourier analysis, and ergodic theory. Notable examples of this connection include communication complexity [16, 9, 14], constructions of randomness extractors [13, 7, 31, 21], and property testing [27, 32]; we also refer the reader to various surveys on additive combinatorics from the vantage point of theoretical computer science [8, 35, 36, 10, 30].

Among the simplest objects of study in additive combinatorics are *sumsets*: A subset S of an abelian group G (with group operation "+") is said to be a *sumset* if $S = A + A$ for some $A \subseteq G$, where for sets $A, B \subseteq G$ we write $A + B$ to denote the set $\{a + b : a \in A, b \in B\}$. Sumsets play a major role in additive combinatorics, where their study has led to many

33rd Annual European Symposium on Algorithms (ESA 2025).
Editors: Anne Benoit, Haim Kaplan, Sebastian Wild, and Grzegorz Herman; Article No. 14; pp. 14:1–14:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

questions and insights about the additive structure of subsets of abelian groups. They are the subject of touchstone results in the field such as Freiman's theorem [23], which (roughly speaking) says that if $|A+A|$ is "not too much larger" than $|A|$ then A must be contained in a generalized arithmetic progression which is "not too large."

Our main interest in this paper is in *algorithmic* questions related to sumsets. In [22] Fagnot, Fertin, and Vialette considered the 2-Sumset Cover problem: given a set S of integers, does there exist a set A of cardinality at most k such that $S \subseteq A+A$? They proved APX-hardness for this problem and presented a $\text{poly}(k) \cdot 5^{k^2(k+3)/2}$-time algorithm. The latter was improved to $\text{poly}(k) \cdot 2^{(3 \log k - 1.4)k}$ by Bulteau, Fertin, Rizzi, and Vialette [15]. 2-Sumset Cover itself specializes Generating Set, in which the goal is to find a minimal set A such that $S \subseteq \{\sum_{i \in I} i \ ; \ I \subseteq A\}$ and which was studied in [17]. Given S and k, finding a set A of size $|A| \geq k$ with $A+A \subseteq S$ is equivalent to finding a k-Clique on the Cayley sum graph of S; this problem remains NP-hard, but can be solved with existing algorithms for k-clique [25]. Recently, Abboud, Fischer, Safier, and Wallheimer proved that recognizing sumsets is NP-complete [1], settling a question raised by Granville [18].

1.1 This Work

In this paper we restrict our attention to the case in which the ambient abelian group G is $\mathbb{F}_2^n$. We do this for several reasons: first, given that our focus is on algorithmic problems, $\mathbb{F}_2^n$ is a very natural domain to consider from a theoretical computer science perspective. Another motivation is that $\mathbb{F}_2^n$ is in some sense the simplest setting for many problems involving sumsets; as Green stated in [27], "the reason that finite field models are nice to work with is that one has the tools of linear algebra, including such notions as subspace and linear independence, which are unavailable in general abelian groups." Indeed, several of our arguments use these linear-algebraic tools.

Since $\mathbb{F}_2^n$ is an exponentially large domain, it is natural to approach the study of sumsets over $\mathbb{F}_2^n$ from the vantage point of *sublinear* algorithms. Thus, we will be interested in algorithms for which either the running time or the number of calls to an oracle for the input set S (i.e. queries of the form "does element x belong to the set S?" is less than 2^n. The recent work [19] took such a sublinear-algorithms perspective; it studied a problem which was closely related to the problem of approximating the size of the sumset $A+A$, given access to an oracle for the unknown set $A \subseteq \mathbb{F}_2^n$. The main result of [19] was that in fact $O_\varepsilon(1)$ queries – in particular, with no dependence on the dimension parameter n – are sufficient for a $\pm\varepsilon \cdot 2^n$-accurate approximation of the quantity that they consider. This naturally motivates the following broad question: What other algorithmic problems involving sumsets may be solvable with "constant" (depending only on ε) or very low query complexity?

Motivated by this general question, in the current work we study a number of algorithmic questions related to sumsets. The main problems we consider are described below:

1. We study (approximate) sumset recognition from a property testing perspective. In more detail, given access to a membership oracle for an unknown set $S \subseteq \mathbb{F}_2^n$, in the *sumset testing problem* the goal is to output "yes" with high probability (say, at least $9/10$) if S is a sumset and "no" with high probability if S is ε-far from every sumset (i.e. $|S \triangle (A+A)| \geq \varepsilon 2^n$ for every set $A \subseteq \mathbb{F}_2^n$), while making as few queries to the oracle as possible.
2. The above-described sumset testing problem turns out to be closely related to the problem of *shift testing*, which is defined as follows: A shift testing algorithm is given black-box access to two oracles $\mathcal{O}_A, \mathcal{O}_B : \mathbb{F}_2^n \to \{0,1\}$, which should be viewed as membership oracles for two subsets $A, B \subseteq \mathbb{F}_2^n$. The algorithm must output "yes" with probability at

least 9/10 if $B = A + \{z\}$ for some string $z \in \mathbb{F}_2^n$ and must output "no" with probability at least 9/10 if the symmetric difference $B \triangle (A + \{z\})$ has size at least $\varepsilon 2^n$ for every $z \in \mathbb{F}_2^n$.

3. For $S \subseteq \mathbb{F}_2^n$, let $\boldsymbol{N}_\varepsilon(S)$ denote a random set which is an "ε-noisy" version of S, obtained by flipping the membership / non-membership of each $x \in \mathbb{F}_2^n$ in S with probability ε. It can be shown that for every $S \subseteq \mathbb{F}_2^n$ and every constant $0 < \varepsilon < 1$, the noisy set $\boldsymbol{N}_\varepsilon(S)$ is with high probability not a sumset. We study the problem of algorithmically *certifying* that $\boldsymbol{N}_\varepsilon(S)$ is not a sumset; i.e. we are given access to a membership oracle for $\boldsymbol{N}_\varepsilon(S)$, where S is an arbitrary and unknown subset of $\mathbb{F}_2^n$, and the goal is to output a set $C \subseteq \mathbb{F}_2^n$ of points such that there is no sumset $A + A$ for which $(A + A) \cap C = \boldsymbol{N}_\varepsilon(S) \cap C$. We refer to this problem as the *smoothed sumset refutation problem,* since it aligns with the well-studied framework of smoothed analysis [34] in which an arbitrary worst-case instance is subjected to a mild perturbation.

The latter is also of non-algorithmic interest, and for completeness we also present (in Section 3) a short proof that the number of different sumsets in $\mathbb{F}_2^n$ is between $2^{2^{n-1}}$ and $2^{2^{n-1}+O(n^2)}$. While an upper bound on the number of such sumsets could have been previously deduced from Theorem 3 in [33], our proof is simpler and gives tighter bounds. Following this work, Alon and Zamir recently further improved these bounds [6].

Our main results are as follows.

Sumset Testing Lower Bound

We give an $\Omega(2^{n/2})$ lower bound on the query complexity of sumset testing:

▶ **Theorem 1.** *There is a constant $\varepsilon > 0$ (independent of n) such that any algorithm $\mathcal{A}$ for the ε-sumset testing problem must make $\Omega(2^{n/2})$ oracle calls.*

Theorem 1 holds even for adaptive testers which may make two-sided error. In particular, note that Theorem 1 rules out the possibility of an efficient tester for the property of being a sumset. Recall that in the property testing literature, "efficient" testers are often defined as algorithms that make a number of queries that depend only on the distance parameter ϵ, or occasionally also polylogarithmic in the problem size. For example, the seminal Blum-Luby-Rubinfeld linearity tester [12] determines if a function $f : \mathbb{F}_2^n \to \mathbb{F}_2$ is linear or ε-far from all linear functions by making $O_\varepsilon(1)$ queries to the function. Other examples include testing for juntas [11], low-degree functions [20], and testing monotonicity [29].

Tight Bounds for Shift Testing

We show that the query complexity of shift testing is $\Theta^*(2^{n/2})$.

▶ **Theorem 2.** *(1) There is an algorithm for the shift testing problem which makes $O(n2^{n/2}/\varepsilon)$ oracle calls and runs in time $\mathrm{poly}(n) \cdot 2^n/\varepsilon$. Moreover, (2) For any constant $0 < c < 1/2$, any algorithm for the shift testing problem must make $\Omega(2^{n/2})$ oracle calls, even for $\varepsilon = (1/2 - 1/2^{cn})$. This lower bound holds even for distinguishing the following two cases: (i) A is a uniform random subset of $\mathbb{F}_2^n$ and $B = A + \{z\}$ for a uniform random $z \in \mathbb{F}_2^n$; versus (ii) A and B are independent uniform random subsets of $\mathbb{F}_2^n$.*

Like Theorem 1, the lower bound, i.e. Part (2), of Theorem 2 holds even for adaptive testers which may make two-sided error.

A Near-Optimal Algorithm for "Smoothed" Sumset Refutation

Our final result is a near-optimal algorithm which certifies that any noisy set $\boldsymbol{N}_\varepsilon(S)$ is not a sumset:

▶ **Theorem 3.** *(1) There is an algorithm for the ε-smoothed sumset refutation problem that makes $2^{n/2} \cdot O(n^{1.5}/\epsilon^{1.5})$ oracle calls and succeeds in certifying that $\boldsymbol{N}_\epsilon(S)$ is not a sumset with probability $1 - o_n(1)$. Moreover, (2) for any constant $\varepsilon > 0$, any algorithm that certifies that $\boldsymbol{N}_\epsilon(S)$ is not a sumset must make $\Omega(2^{n/2}/\sqrt{n})$ many oracle calls.*

1.2 Technical Overview

The $\mathbb{F}_2^n$ testing setting allows us to employ algorithms that are conceptually straightforward, even if proving correctness requires some care.

The main idea of our algorithm for shift testing (Part (1) of Theorem 2) is to query one oracle with all shifts of a random point r by a subspace V, and query the other oracle by all shifts of the same point r by the orthogonal complement $V^\perp$. This requires only $O(2^{n/2})$ queries, while providing information about the relationship between the two oracles vis-a-vis any possible shift $z \in \mathbb{F}_2^n$, since every possible shift has a decomposition into $z = z_1 + z_2$ for some $z_1 \in V, z_2 \in V^\perp$.

The optimality of this general approach is witnessed by the lower bound in Part (2) of Theorem 2. The proof is by a "deferred decisions" argument which analyzes the knowledge transcript of a query algorithm which may be interacting either with the "yes"-pair of oracles or the "no"-pair of oracles. We describe a coupling of the knowledge transcripts between these two cases, and use it to argue that if fewer than $0.1 \cdot 2^{n/2}$ queries have been made, then with high probability the transcripts are identically distributed across these two cases. (See [26] for a similar high-level argument, though in an entirely different technical setting.)

The $\Omega(2^{n/2})$ lower bound of Theorem 1 for sumset testing is by a reduction to the lower bound for shift testing. We give a straightforward embedding of the "A is random, $B = A + \{z\}$"-versus-"A, B are independent random" shift testing problem over $\mathbb{F}_2^n$ into the problem of sumset testing over $\mathbb{F}_2^{n+2}$. The most challenging part of the argument is to prove that in fact the "no" instances of shift testing (when A, B are independent random sets) give rise to instances which are far from sumsets over $\mathbb{F}_2^{n+2}$. This requires us to argue that a subset of $\mathbb{F}_2^{n+2}$ which is constant on two n-dimensional cosets and is uniform random on the other two n-dimensional cosets, is likely to be far from every sumset, which we prove using a linear algebraic argument.

For Theorem 3, a result due to Alon establishes that every subset of the Boolean cube of size $2^n - c2^{n/2}/\sqrt{n}$ is a sumset for a small constant c, which implies that any sumset "0-certificate" has size $\Omega^*(2^{n/2})$ [3]. To find such a certificate, we show that few noisy sumsets are likely to be consistent with an arbitrarily chosen subspace of dimension $n/2$, and then use a small random sample to rule out these sumsets with high probability.

1.3 Discussion

Our results suggest many questions and goals for future work; we record two such directions here.

The first direction is to obtain stronger results on sumset refutation. Is it possible to strengthen our sumset refutation result by eliminating the "smoothed analysis" aspect, i.e. is it the case that any $S \subseteq \mathbb{F}_2^n$ that is ε-far from every sumset has a "0-certificate" of size $2^{n/2} \cdot \mathrm{poly}\left(n, \frac{1}{\varepsilon}\right)$? If so, can such certificates be found efficiently given query access to S?

The second, and perhaps most compelling, direction is to either strengthen our $\Omega(2^{n/2})$-query lower bound, or prove an upper bound, for the sumset testing problem. We are cautiously optimistic that the true query complexity of sumset testing may be closer to $2^{n/2}$ queries than to 2^n queries, but any nontrivial ($o(2^n)$-query) algorithm would be an interesting result. One potentially relevant intermediate problem towards sumset testing is the problem of *k-shift testing*, in which the goal is to determine whether oracles $\mathcal{O}_A, \mathcal{O}_B : \mathbb{F}_2^n \to \{0,1\}$ correspond to $B = A + \{s_1, \dots, s_k\}$ for some k "shift" vectors $(s_i)_{i\in[k]}$ versus B being ε-far from every union of k shifts of A.

2 Preliminaries

All probabilities and expectations will be with respect to the uniform distribution, unless otherwise indicated. We use boldfaced characters such as $\boldsymbol{x}$, $\boldsymbol{f}$, and $\boldsymbol{A}$ to denote random variables (which may be real-valued, vector-valued, function-valued, or set-valued; the intended type will be clear from the context). We write $\boldsymbol{x} \sim \mathcal{D}$ to indicate that the random variable $\boldsymbol{x}$ is distributed according to the probability distribution $\mathcal{D}$. We write $\mathrm{dist}_{\mathrm{TV}}(\mathcal{D}_1, \mathcal{D}_2)$ to denote the *total variation distance* or *statistical distance* between the distributions $\mathcal{D}_1$ and $\mathcal{D}_2$.

For $\epsilon \in [0,1]$, we write $\boldsymbol{R}_\varepsilon$ to denote a random subset of $\mathbb{F}_2^n$ obtained by selecting each element with probability ϵ, so the "ε-noisy version" of a set $S \subseteq \mathbb{F}_2^n$, denoted $\boldsymbol{N}_\varepsilon(S)$, is equivalent to $S \triangle \boldsymbol{R}_\varepsilon$, where $A \triangle B := (A \setminus B) \cup (B \setminus A)$ denotes the symmetric difference of A and B.

Given a set $A \subseteq \mathbb{F}_2^n$, we will write $\mathcal{O}_A : \mathbb{F}_2^n \to \{0,1\}$ to denote the membership oracle for A, i.e.

$$\mathcal{O}_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases}$$

for $x \in \mathbb{F}_2^n$. Given $A, B \subseteq \mathbb{F}_2^n$, we write $\mathrm{dist}(A,B)$ for the normalized Hamming distance between the sets A and B, i.e.

$$\mathrm{dist}(A,B) := \frac{|A \triangle B|}{2^n} = \Pr_{\boldsymbol{x}\sim\mathbb{F}_2^n}[\mathcal{O}_A(\boldsymbol{x}) \neq \mathcal{O}_B(\boldsymbol{x})].$$

We will also write $A + B := \{a + b : a \in A, b \in B\}$. If one of the sets is a singleton, e.g. if $A = \{a\}$, we will sometimes write $a + B := \{a\} + B$ instead.

We write $H(x)$ to denote the binary entropy function $-x\log_2 x - (1-x)\log_2(1-x)$. Stirling's approximation gives us the following helpful identity:

$$\binom{n}{\alpha n} = \Theta^*(2^{H(\alpha)n}); \text{ or, equivalently, } \binom{2^n}{\alpha 2^n} = 2^{H(\alpha)2^n} \cdot 2^{\Theta(n)}. \tag{1}$$

Given a subset D of an Abelian group G, we write $\Gamma_G(D)$ to denote the *Cayley sum graph* of G with respect to the generator set D; that is, the graph on the vertex set G that contains the edge (x,y) if and only if $x + y \in D$. (Since the group we consider is $\mathbb{F}_2^n$, for us this is the same as the regular Cayley graph of G with respect to generator set D.) When $D = \{x\}$ is a singleton for some $x \in G$, we abuse notation slightly and write $\Gamma_G(x)$ for $\Gamma_G(\{x\})$.

3 The Number of Sumsets in $\mathbb{F}_2^n$

▶ **Proposition 4.** *The number of sumsets in $\mathbb{F}_2^n$ is at most*

$$2^{2^{n-1}+O(n^2)}.$$

Proof. Consider a sumset $S = A + A$; we consider two cases depending on the linear rank of the set $\mathbb{F}_2^n \setminus S$.

Case 1: $\mathbb{F}_2^n \setminus S$ does not have full rank. In other words, there exists a vector $v \in \mathbb{F}_2^n$ such that

$$\langle x, v\rangle = 1 \qquad \text{implies that} \qquad x \in S.$$

In particular, we have that $S = S' \cup \left(\mathbb{F}_2^n \setminus v^\perp\right)$ for some $S' \subseteq v^\perp = \{x \in \mathbb{F}_2^n : \langle x, v\rangle = 0\}$. As there are at most 2^n choices for v, and for each choice of v there are at most $2^{2^{n-1}}$ choices for S', we have that there are at most $2^{2^{n-1}+n}$ many sumsets of this form.

Case 2: $\mathbb{F}_2^n \setminus S$ has full rank. In particular, there are n linearly independent vectors *not* in S. For $v \in \mathbb{F}_2^n$, observe that the Cayley graph $\Gamma_{\mathbb{F}_2^n}(v)$ is a perfect matching. Next, note that if $v \notin S = A + A$, then A must be an independent set in $\Gamma_{\mathbb{F}_2^n}(v)$. This is because, if $x, y \in A$ with $x + v = y$ then

$$A + A \ni x + y = x + x + v = v \notin A + A,$$

which is a contradiction. As we have n linearly independent vectors not in S, it follows that there exists an orthogonal transformation of $\mathbb{F}_2^n$ such that A must be an independent set in the hypercube (where edges are incident to elements of $\mathbb{F}_2^n$ that differ in a single coordinate). As the number of independent sets in the hypercube Q_n is at most $2^{2^{n-1}+O(1)}$ (see for example [24]), and as the number of orthogonal transformations of the hypercube is at most 2^{n^2}, it follows that the total number of sumsets of this form is at most

$$2^{2^{n-1}+n^2+O(1)}.$$

Both cases together complete the proof. ◀

▶ **Proposition 5.** *The number of sumsets in $\mathbb{F}_2^n$ is at least $2^{2^{n-1}}$.*

Proof. For any subset $A \subseteq \mathbb{F}_2^{n-1}$ of the $(n-1)$-th dimensional hypercube, we define a subset $A' \subseteq \mathbb{F}_2^n$ as

$$A' := \{\vec{0}\} \cup \{(1, a) \mid a \in A\},$$

where for a $(n-1)$-dimensional vector a, the concatenation $(1, a)$ is defined as the n-dimensional vector where the first coordinate is 1 and the other $(n-1)$ coordinates are equal to a. We observe that $(A' + A') \cap \left(\mathbb{F}_2^n \setminus e_1^\perp\right) = \{(1, a) \mid a \in A\}$. That is, in the sumset $(A' + A')$ all vectors in which the first coordinate is 1 exactly correspond to the set A. In particular, for any $A_1 \neq A_2 \in \mathbb{F}_2^{n-1}$, we have $(A_1' + A_1') \neq (A_2' + A_2')$. ◀

4 Optimally Testing Shifts

Given $A, B \subseteq \mathbb{F}_2^n$, we say that B is a *shift* of A if there exists $z \in \mathbb{F}_2^n$ such that $A + z = B$. We obtain the following upper and lower bounds for the *shift testing* problem:

▶ **Theorem 6.** *Let $\mathcal{O}_A, \mathcal{O}_B : \mathbb{F}_2^n \to \{0, 1\}$ be membership oracles for $A, B \subseteq \mathbb{F}_2^n$. Then:*

1. *The algorithm* Shift-Tester *(Algorithm 1) makes $O(n2^{n/2}/\varepsilon)$ oracle calls, runs in time $\mathrm{poly}(n) \cdot 2^n/\varepsilon$, and guarantees that:*
   - **a.** *If $B = A + z$ for some $z \in \mathbb{F}_2^n$, the algorithm outputs "shift" with probability $9/10$;*
   - **b.** *If for every $z \in \mathbb{F}_2^n$ we have $\mathrm{dist}(A + z, B) \geq \varepsilon$, the algorithm outputs "ε-far from shift" with probability $9/10$.*

2. *Fix c to be a constant that is less than $1/2$. Any (adaptive, randomized) algorithm with the performance guarantee in the previous item makes $\Omega(2^{n/2})$ oracle calls, even for $\varepsilon = 1/2 - 1/2^{cn}$.*

In fact, the lower bound holds even for distinguishing the following two cases: (i) A is a uniform random subset of $\mathbb{F}_2^n$ and $B = A + s$ for a uniform random $s \in \mathbb{F}_2^n$; versus (ii) A and B are independent uniform random subsets of $\mathbb{F}_2^n$.

4.1 Upper Bound

In this section, we prove Item 1 of Theorem 6. Note that since

$$\mathrm{dist}(B, A+z) = \Pr_{\boldsymbol{x}}\left[\mathcal{O}_A(\boldsymbol{x}) \neq \mathcal{O}_B(\boldsymbol{x}+z)\right],$$

if B is a shift of A (i.e. $B = A + z_*$ for some z_*), we then have for that z_* that

$$\Pr_{\boldsymbol{x}}\left[\mathcal{O}_B(\boldsymbol{x}) = \mathcal{O}_A(\boldsymbol{x}+z_*)\right] = 1.$$

On the other hand, if $\mathrm{dist}(B, A+z) \geq \varepsilon$ for every z, then for every z we have

$$\Pr_{\boldsymbol{x}}\left[\mathcal{O}_B(\boldsymbol{x}) = \mathcal{O}_A(\boldsymbol{x}+z)\right] \leq 1 - \varepsilon.$$

These simple observations suggest that in order to estimate $\Pr[\mathcal{O}_B(\boldsymbol{x}) = \mathcal{O}_A(\boldsymbol{x}+z)]$ for a particular z, we would like to make queries $\mathcal{O}_B(\boldsymbol{x}), \mathcal{O}_A(\boldsymbol{x}+z)$ for uniform random $\boldsymbol{x}$. The fact that we need to do this for all z motivates the following approach; before proceeding, we introduce some notation.

▶ **Notation 7.** *We define the subsets $D_1, D_2 \subset \mathbb{F}_2^n$, where D_1 is the set of all $2^{\lfloor n/2 \rfloor}$ vectors whose last $\lfloor n/2 \rfloor$ coordinates are all-0 and $D_2 \subset \mathbb{F}_2^n$ is the set of $2^{\lceil n/2 \rceil}$ vectors whose first $\lceil n/2 \rceil$ coordinates are all-0. Note that every $z \in \mathbb{F}_2^n$ has a unique expression as*

$$z := z^{(1)} + z^{(2)}, \qquad \textit{for } z^{(1)} \in D_1 \textit{ and } z^{(2)} \in D_2.$$

Fix a particular string $z = z^{(1)} + z^{(2)}$ as above. We write $\boldsymbol{x} = \boldsymbol{r} + z^{(1)}$, and we observe that if $\boldsymbol{r}$ is uniform random then so is $\boldsymbol{x}$. As alluded to earlier we would like to query B on $\boldsymbol{x}$ and A on $\boldsymbol{x} + z = \boldsymbol{r} + z^{(1)} + z = \boldsymbol{r} + z^{(2)}$. The main observation is that if we query B on all strings in $D_1 + \boldsymbol{r}$ and query A on all strings in $D_2 + \boldsymbol{r}$, then no matter what z is we will have made the queries $\mathcal{O}_B(\boldsymbol{x}) = \mathcal{O}_B(\boldsymbol{r} + z^{(1)})$ and $\mathcal{O}_A(\boldsymbol{x}+z) = \mathcal{O}_A(\boldsymbol{r} + z^{(2)})$, so we will have obtained a sample towards estimating $\Pr_{\boldsymbol{x}}[\mathcal{O}_B(\boldsymbol{x}) = \mathcal{O}_A(\boldsymbol{x}+z)]$. Since this is true for every z, we can reuse the above queries towards all possibilities for z. (Of course one sample is not enough to estimate a probability, so we will repeat the above with n/ε different choices of $\boldsymbol{r}$.)

Proof of Item 1 of Theorem 6. Our algorithm, Shift-Tester, is presented in Algorithm 1. Note that if B is a shift of A, i.e. if there exists a $z_* \in \mathbb{F}_2^n$ for which $B = A + z_*$ then

$$\boldsymbol{r} + z_*^{(1)} \in A \qquad \text{if and only if} \qquad \boldsymbol{r} + z_*^{(1)} + z_* = \boldsymbol{r} + z_*^{(2)} \in B,$$

where we used the fact that $z_*^{(1)} + z_* = z_*^{(1)} + z_*^{(1)} + z_*^{(2)} = z_*^{(2)}$. In particular, we will have $p_{z_*} = 1$ and so the algorithm will return "shift" with probability 1. On the other hand, suppose B is ε-far from $A + z$ for every $z \in \mathbb{F}_2^n$; fix any such z. Then the probability that all n/ε repetitions in Algorithm 1 will have $\mathcal{O}_A(\boldsymbol{r} + z^{(1)}) = \mathcal{O}_B(\boldsymbol{r} + z^{(2)})$ is at most

$$(1-\varepsilon)^{n/\varepsilon} \leq e^{-n}.$$

Taking a union bound over all $z \in \mathbb{F}_2^n$ implies that the probability that Algorithm 1 will output "ε-far from shift" is at least $1 - (2/e)^n$, completing the proof. ◀

Algorithm 1 An algorithm for shift testing.

```
Input: Oracles $\mathcal{O}_A, \mathcal{O}_B : \mathbb{F}_2^n \to \{0,1\}$ and $\epsilon > 0$
Output: “Shift” or “$\varepsilon$-far from shift”

SHIFT-TESTER($\mathcal{O}_A, \mathcal{O}_B, \epsilon$):
1. Repeat the following $n/\epsilon$ times:
    a. Draw a uniformly random $\boldsymbol{r} \in \mathbb{F}_2^n$.
    b. Query $\mathcal{O}_A$ on all $x \in D_1 + \boldsymbol{r}$, and query $\mathcal{O}_B$ on all $y \in D_2 + \boldsymbol{r}$.
2. For each $z = z^{(1)} + z^{(2)} \in \mathbb{F}_2^n$, let $p_z$ be the fraction of the $n/\varepsilon$ iterations for which
        $\mathcal{O}_A(\boldsymbol{r} + z^{(1)}) = \mathcal{O}_B(\boldsymbol{r} + z^{(2)})$.
3. If $p_z = 1$ for some $z \in \mathbb{F}_2^n$, output “shift”; otherwise output “$\varepsilon$-far from shift”.
```

Note that Algorithm 1 in fact guarantees 1-sided error, stronger than what is required by Theorem 6: The algorithm never outputs “ε-far from shift” if B is a shift of A, and if B is ε-far from every shift of A then the algorithm outputs “shift” with probability at most $(2/e)^n$.

4.2 Lower Bound

To prove Item 2 of Theorem 6 we define two probability distributions, $\mathcal{D}_{\text{yes}}$ and $\mathcal{D}_{\text{no}}$, over instances of the shift testing problem.

▶ **Definition 8.** *A draw $(\boldsymbol{A}, \boldsymbol{B})$ from $\mathcal{D}_{\text{yes}}$ is obtained as follows:*

- $\boldsymbol{A} \subseteq \mathbb{F}_2^n$ *includes each element of $\mathbb{F}_2^n$ independently with probability 1/2.*
- $\boldsymbol{B} \subseteq \mathbb{F}_2^n$ *equals $\boldsymbol{A} + \boldsymbol{s}$ for $\boldsymbol{s}$ sampled uniformly at random from $\mathbb{F}_2^n$.*

Note that for $(\boldsymbol{A}, \boldsymbol{B}) \sim \mathcal{D}_{\text{yes}}$, $\boldsymbol{B}$ is a shift of $\boldsymbol{A}$.

▶ **Definition 9.** *A draw $(\boldsymbol{A}, \boldsymbol{B})$ from $\mathcal{D}_{\text{no}}$ is obtained as follows:*

- $\boldsymbol{A} \subseteq \mathbb{F}_2^n$ *includes each element of $\mathbb{F}_2^n$ independently with probability 1/2.*
- $\boldsymbol{B} \subseteq \mathbb{F}_2^n$ *also includes each element of $\mathbb{F}_2^n$ with probability 1/2 (independently of $\boldsymbol{A}$).*

A straightforward application of the Chernoff bound, combined with a union bound over the 2^n possible shifts, shows that with probability at least 19/20 a draw of $(\boldsymbol{A}, \boldsymbol{B}) \sim \mathcal{D}_{\text{no}}$ is such that $\boldsymbol{B}$ is $(1/2 - 1/2^{cn})$-far from every shift of $\boldsymbol{A}$ (for any constant $c < 1/2$). So to prove Item 2 of Theorem 2, it is enough to establish the following claim for deterministic algorithms. (By Yao's minimax principle, this is sufficient to prove a lower bound for randomized algorithms as well.)

▷ Claim 10. Let `Test` be any deterministic, adaptive algorithm that makes $N := 0.1 \cdot 2^{n/2}$ oracle calls to $\mathcal{O}_A$ and $\mathcal{O}_B$. Let $T_{\texttt{test}}(A, B)$ be the “transcript” of its queries to the oracles and received responses, i.e. $T_{\texttt{test}}(A, B)$ consists of

$$\begin{gathered}\text{(first query to one of the oracles, response received)}\\ \vdots \\ \text{(N-th query to one of the oracles, response received).}\end{gathered}$$

Then we have

$$\mathrm{dist}_{\mathrm{TV}}\left(T_{\texttt{test}}(\boldsymbol{A}_{\text{yes}}, \boldsymbol{B}_{\text{yes}}), T_{\texttt{test}}(\boldsymbol{A}_{\text{no}}, \boldsymbol{B}_{\text{no}})\right) \leq 0.02,$$

where $(\boldsymbol{A}_{\text{yes}}, \boldsymbol{B}_{\text{yes}}) \sim \mathcal{D}_{\text{yes}}$ and $(\boldsymbol{A}_{\text{no}}, \boldsymbol{B}_{\text{no}}) \sim \mathcal{D}_{\text{no}}$.

The claim follows by analyzing the behavior of the algorithm on an oracle constructed over the course of answering the queries posed by algorithm `Test`, i.e., "deferring" the decision of whether the oracle $(\boldsymbol{A}, \boldsymbol{B})$ is drawn from $\mathcal{D}_{\text{yes}}$ or $\mathcal{D}_{\text{no}}$. See for example Section 7.1 of [26].

Proof. For simplicity, we assume that in each round, `Test` queries one point q and receives *both* $\mathcal{O}_A(q)$ and $\mathcal{O}_B(q)$; this can only make `Test` more powerful.

Consider the following approach to answering queries posed by `Test`: before any queries are made, draw a uniform random $\boldsymbol{s} \sim \mathbb{F}_2^n$. Let $q_1, \ldots, q_{t-1} \in \mathbb{F}_2^n$ be the first $t-1$ queries made by `Test` (we may suppose without loss of generality that all these $t-1$ query strings are different from each other, since any algorithm that repeats a query string can easily be modified so as not to do so). When the t-th query string q_t is provided by `Test`, the answer is generated as follows:

1. If $\boldsymbol{s} \neq q_t + q_{t'}$ for all $t' \leq t$, then two independent uniform random bits $\boldsymbol{b_A}, \boldsymbol{b_B} \in \{0, 1\}$ are drawn and returned as $\mathcal{O}_{\boldsymbol{A}}(q_t)$ and $\mathcal{O}_{\boldsymbol{B}}(q_t)$. (It may be helpful to think of this outcome as being "recorded", i.e. when this happens the process "decides" that $\boldsymbol{b_A}, \boldsymbol{b_B}$ are the values of $\boldsymbol{A}$ and $\boldsymbol{B}$ on the point q_t.)
2. If $\boldsymbol{s} = q_t + q_{t'}$ for some $t' \leq t$, then the process halts and outputs "failure."

The key observation is that conditioned on the above process proceeding through t queries without an output of "failure", the length-t transcript is distributed exactly according to the pair of oracles $(\boldsymbol{A}, \boldsymbol{B})$ being $(\boldsymbol{A}_{\text{yes}}, \boldsymbol{B}_{\text{yes}}) \sim \mathcal{D}_{\text{yes}}$, and also exactly according to the pair of oracles being $(\boldsymbol{A}_{\text{no}}, \boldsymbol{B}_{\text{no}}) \sim \mathcal{D}_{\text{no}}$. This is because in either case, as long as no pair of queries $q_t, q_{t'}$ sum to the "hidden" random string $\boldsymbol{s} \in \mathbb{F}_2^n$, every response to every oracle call is distributed as an independent uniform random bit.

We finish the proof by showing that the probability that the process above outputs "failure" is at most 0.02. We emphasize that the probability here is taken over the entire random process which includes the initial uniform random draw of $\boldsymbol{s} \sim \mathbb{F}_2^n$.

To this end, we note that conditioning on no "failure" during the first $t-1$ rounds $q_1, \ldots, q_{t-1}$, $\boldsymbol{s}$ is distributed uniformly among all points in $\mathbb{F}_2^n$ that are not equal to $q_i + q_j$ for some $i, j \in [t-1]$. The number of such points is at least $2^n - N^2/2 > 0.99 \cdot 2^n$. On the other hand, the process outputs "failure" in round t if one of $q_1 + q_t, \ldots, q_{t-1} + q_t$ is $\boldsymbol{s}$, which happens with probability at most $N/(0.99 \cdot 2^n) < 0.2 \cdot 2^{-n/2}$. It follows from a union bound on the N rounds that the process outputs "failure" with probability at most 0.02. This finishes the proof of the claim. ◁

5 Lower Bound for Testing Sumsets

In this section, we show that the lower bound for shift testing established in Section 4.2 implies a lower bound for the problem of testing sumsets. More formally, we prove the following:

▶ **Theorem 11.** *Let $\mathcal{O}_S : \mathbb{F}_2^n \to \{0, 1\}$ be a membership oracle for $S \subseteq \mathbb{F}_2^n$. There is an absolute constant $\varepsilon > 0.0125$ such that the following holds: Let $\mathcal{A}$ be any (adaptive, randomized) algorithm with the following performance guarantee:*

1. *If $S = A + A$ for some $A \subseteq \mathbb{F}_2^n$, $\mathcal{A}$ outputs "sumset" with probability 9/10; and*
2. *If $\operatorname{dist}(S, A + A) \geq \epsilon$ for all $A \subseteq \mathbb{F}_2^n$, $\mathcal{A}$ outputs "ε-far from sumset" with probability 9/10.*

Then $\mathcal{A}$ must make $\Omega(2^{n/2})$ calls to $\mathcal{O}_S$.

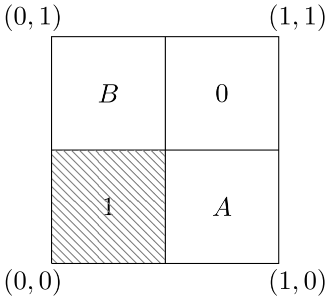

Figure 1 The set $\mathcal{S}(A, B) \subseteq \mathbb{F}_2^{n+2}$. By Proposition 12, for a typical $(\boldsymbol{A}_{\text{yes}}, \boldsymbol{B}_{\text{yes}})$ drawn from $\mathcal{D}_{\text{yes}}$, adding a single point $(1, 1, \boldsymbol{s})$ in the top right cell makes $S(\boldsymbol{A}_{\text{yes}}, \boldsymbol{B}_{\text{yes}})$ into a sumset.

The distributions we use to prove Theorem 11 are based on the distributions $\mathcal{D}_{\text{yes}}$ and $\mathcal{D}_{\text{no}}$ defined in Definitions 8 and 9 for shift testing. Given $A, B \subseteq \mathbb{F}_2^n$, we define $\mathcal{S}(A, B) \subseteq \mathbb{F}_2^{n+2}$ as

$$\mathcal{S}(A, B) := \{x : x_1 = x_2 = 0\} \sqcup \{(1, 0, a) : a \in A\} \sqcup \{(0, 1, b) : b \in B\}, \tag{2}$$

where the notation (b_1, b_2, v) indicates that the bits b_1 and b_2 are concatenated with $v \in \mathbb{F}_2^n$ to create an element in $\mathbb{F}_2^{n+2}$. Figure 1 illustrates the set $\mathcal{S}(A, B)$.

We use $\mathcal{D}_{\text{no}}$ to define $\mathcal{S}_{\text{no}}$, a distribution over subsets of $\mathbb{F}_2^{n+2}$ as follows: To draw $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$, we draw $(\boldsymbol{A}_{\text{no}}, \boldsymbol{B}_{\text{no}}) \sim \mathcal{D}_{\text{no}}$ and set $\boldsymbol{S}_{\text{no}} = \mathcal{S}(\boldsymbol{A}_{\text{no}}, \boldsymbol{B}_{\text{no}})$. On the other hand, we use $\mathcal{D}_{\text{yes}}$ to define $\mathcal{S}_{\text{yes}}$ as follows: To draw $\boldsymbol{S}_{\text{yes}} \sim \mathcal{S}_{\text{yes}}$, we draw $(\boldsymbol{A}_{\text{yes}}, \boldsymbol{B}_{\text{yes}} = \boldsymbol{A}_{\text{yes}} + \boldsymbol{s}) \sim \mathcal{D}_{\text{yes}}$ but add one "extra" point to $\boldsymbol{S}_{\text{yes}}$, defining it as: $\boldsymbol{S}_{\text{yes}} = \mathcal{S}(\boldsymbol{A}_{\text{yes}}, \boldsymbol{B}_{\text{yes}}) \sqcup \{(1, 1, \boldsymbol{s})\}$. This will ensure that $\boldsymbol{S}_{\text{yes}} \sim \mathcal{S}_{\text{yes}}$ is likely to be a sumset (see Proposition 12 below).

At a high level, the proof of Theorem 11 contains three steps: we show (1) that $\boldsymbol{S}_{\text{yes}} \sim \mathcal{S}_{\text{yes}}$ is a sumset with high probability (Proposition 12), (2) that $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$ is ϵ-far from being a sumset with high probability (Proposition 13), and (3) that oracles to $\boldsymbol{S}_{\text{yes}} \sim \mathcal{S}_{\text{yes}}$ and $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$ are too similar for an algorithm that makes few queries to tell the difference, where "similarity" is measured in terms of the total variation distance between distributions over transcripts (proof of Theorem 11). The theorem then follows quickly from these three facts.

▶ **Proposition 12.** *With probability at least $1 - 2^{-\Omega(2^n)}$, $\boldsymbol{S}_{\text{yes}} \sim \mathcal{S}_{\text{yes}}$ is a sumset over $\mathbb{F}_2^{n+2}$.*

Proof. Let $(A_{\text{yes}}, B_{\text{yes}})$ be a pair of sets in the support of $\mathcal{D}_{\text{yes}}$ with $B_{\text{yes}} = A_{\text{yes}} + s$. It is easy to verify that $\mathcal{S}(A_{\text{yes}}, B_{\text{yes}}) \sqcup \{(1, 1, s)\}$ is equal to $C + C$ with

$$C := \{0^n\} \sqcup \{(1, 0, a) : a \in A_{\text{yes}}\} \sqcup \{(1, 1, s)\}\,.$$

as long as $A_{\text{yes}} + A_{\text{yes}}$ covers all of $\mathbb{F}_2^n$. So it suffices to show that this holds with extremely high probability with a uniformly random set $\boldsymbol{A}_{\text{yes}}$.

To see this, consider any fixed, nonzero element $z \in \mathbb{F}_2^n$. Without loss of generality, suppose that the first coordinate of z is 1. We have

$$\Pr\left[z \notin \boldsymbol{A}_{\text{yes}} + \boldsymbol{A}_{\text{yes}}\right] = \Pr\left[\text{for all } y \in \mathbb{F}_2^n,\ \text{either } y \notin \boldsymbol{A}_{\text{yes}} \text{ or } z + y \notin \boldsymbol{A}_{\text{yes}}\right] = (3/4)^{2^{n-1}},$$

where the second equality holds because $\boldsymbol{A}_{\text{yes}}$ is a uniform random subset of $\mathbb{F}_2^n$ and $y, z + y$ are distinct elements (observe that the first coordinate of y is 0 while the first coordinate of $z + y$ is 1). Since $\Pr[0^n \notin \boldsymbol{A}_{\text{yes}} + \boldsymbol{A}_{\text{yes}}] = \Pr[\boldsymbol{A}_{\text{yes}} \text{ is empty}] = (1/2)^{2^n} < (3/4)^{2^{n-1}}$, we get that each fixed element $z \in \mathbb{F}_2^n$ is missing from $\boldsymbol{A}_{\text{yes}} + \boldsymbol{A}_{\text{yes}}$ with probability at most $(3/4)^{2^{n-1}}$. The claim follows from a union bound over the 2^n elements of $\mathbb{F}_2^n$. ◀

▶ **Proposition 13.** *With probability at least* $1 - o_n(1)$*,* $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$ *is* 0.0125*-far from every sumset.*

We now complete the proof of Theorem 11 using Propositions 12 and 13. The proof of Proposition 13 is deferred to Section 5.1.

Proof of Theorem 11. Let $\mathcal{A}$ be an algorithm for sumset testing on $\mathbb{F}_2^{n+2}$ that makes at most $N = 0.1 \cdot 2^{n/2}$ queries. As in the proof of Theorem 6, we let $T_{\mathcal{A}}(S)$ denote the N-element transcript of $\mathcal{A}$ given the oracle $\mathcal{O}_S$ and take a "deferred decision" approach to prove that $\mathcal{A}$ cannot distinguish between $\mathcal{S}_{\text{yes}}$ and $\mathcal{S}_{\text{no}}$ with high probability.

By Propositions 12 and 13, the probability that $\boldsymbol{S}_{\text{yes}} \sim \mathcal{S}_{\text{yes}}$ is not a sumset is $o_n(1)$, and the probability that $\boldsymbol{S} \sim \mathcal{S}_{\text{no}}$ is 0.0125-close to any sumset is $o_n(1)$. As a result, to prove Theorem 11 it suffices to show that

$$\text{dist}_{\text{TV}}\left(T_{\mathcal{A}}\big(\boldsymbol{S}_{\text{yes}}\big), T_{\mathcal{A}}(\boldsymbol{S}_{\text{no}})\right) < 0.1 - o_n(1),$$

where $\boldsymbol{S}_{\text{yes}} \sim \mathcal{S}_{\text{yes}}$ and $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$.

Consider the sham oracle $\mathcal{O}_{\text{sham}}$ that samples a point $\boldsymbol{s} \in \mathbb{F}_2^n$ uniformly at random, then responds to queries as follows:

1. If the query is a point q_t for which $q_{t,1} = q_{t,2} = 0$, the oracle returns 1.
2. If the query is $(1, 1, \boldsymbol{s})$, the oracle outputs "failure". Otherwise, if $q_{t,1} = q_{t,2} = 1$, it returns 0.
3. If the query is a point q_t such that $q_t + q_{t'} = (1, 1, \boldsymbol{s})$ for some previously queried point $q_{t'}$, the oracle outputs "failure". Otherwise, it returns a random bit.

We proceed to consider the behavior of $\mathcal{A}$ given $\mathcal{O}_{\text{sham}}$, $\mathcal{O}_{\boldsymbol{S}_{\text{yes}}}$, and $\mathcal{O}_{\boldsymbol{S}_{\text{no}}}$. Conditioned on the event that $\mathcal{O}_{sham}$ does not output "failure", $\mathcal{A}$ always receives the answer '1' when querying a point with initial coordinates $(0, 0)$, always receives the answer '0' when querying a point with initial coordinates $(1, 1)$, and receives a random bit when querying a point with the initial coordinates $(0, 1)$ or $(1, 0)$. If, after the point of "failure", our oracle subsequently responds to queries consistently with the distribution $\mathcal{S}(\boldsymbol{A}_{\text{yes}}, \boldsymbol{A}_{\text{yes}} + \boldsymbol{s})$, randomly determining membership in $\boldsymbol{A}_{\text{yes}}$ via deferred decision as necessary, the resulting distribution over transcripts is identical to that given oracle access to $\boldsymbol{S}_{\text{yes}}$. Likewise, if the oracle responds '0' on $(1, 1, s)$ and continues to return random bits on queries whose initial coordinates begin with $(0, 1)$ or $(1, 0)$, the resulting distribution over transcripts is identical to that given oracle access to $\boldsymbol{S}_{\text{no}}$. We conclude that the distribution of $T_{\mathcal{A}}(\text{sham})$, the transcript of $\mathcal{A}$ given $\mathcal{O}_{\text{sham}}$, is identical to the distribution of transcripts given $\mathcal{O}_{\boldsymbol{S}_{\text{yes}}}$ and $\mathcal{O}_{\boldsymbol{S}_{\text{no}}}$ unless failure occurs.

Failure is unlikely for any algorithm $\mathcal{A}$ that makes at most N queries: With N queries, the algorithm can rule out at most $N = O(2^{n/2})$ candidates for $\boldsymbol{s}$ by querying points with the initial coordinates $(1, 1)$, and at most $N^2 = 0.01 \cdot 2^n$ candidates for $\boldsymbol{s}$ by querying points with the initial coordinates $(0, 1)$ and $(1, 0)$. Conditioned on no failure, the posterior distribution of $\boldsymbol{s}$ is thus uniform over at least $(0.99 - o_n(1))2^n$ points. Thus subsequently querying a point discovers s with probability at most

$$\frac{N}{(0.99 - o_n(1))2^n} \le \frac{0.2}{2^{n/2}}.$$

Union-bounding over all N rounds gives a failure probability of at most 0.02.

We conclude that

$$\text{dist}_{\text{TV}}(T_{\mathcal{A}}(\boldsymbol{S}_{\text{yes}}), T_{\mathcal{A}}(\boldsymbol{S}_{\text{no}})) \le 0.02 + o_n(1),$$

and thus any algorithm that makes at most $N = 0.1 \cdot 2^{n/2}$ queries cannot answer correctly with probability 9/10. ◀

5.1 Proof of Proposition 13

We prove Proposition 13 via a counting argument. The distribution $\mathcal{S}_{\text{no}}$ produces subsets of $\mathbb{F}_2^{n+2}$ of a specific form: these subsets contain every point in the subspace $\{x : x_1 = x_2 = 0\}$, no points in the coset $\{x : x_1 = x_2 = 1\}$, and have density roughly 0.5 on the cosets $\{x : x_1 = 0, x_2 = 1\}$ and $\{x : x_1 = 1, x_2 = 0\}$. We first bound the number of sumsets that are ϵ-*eligible* (roughly, "close") to any subset of this form (Proposition 16). Since there are relatively few subsets of $\mathbb{F}_2^{n+2}$ near any ϵ-eligible sumset, we conclude that most subsets drawn from $\mathcal{S}_{\text{no}}$ are far from any sumset (Proposition 13).

▶ Remark 14. It can be shown that the number of sumsets in $\mathbb{F}_2^{n+2}$ is at most $2^{2^{n+1}+O(n^2)}$ (this bound is implicit in the work [33], and for completeness we give a proof in Section 3). However, this upper bound is not enough for us per se since the support of $\mathcal{S}_{\text{no}}$ is also of size $2^{2^{n+1}}$; hence we need to use the more refined notion of "ε-eligible" sumsets mentioned above.

In the remainder of this section, we make frequent reference to the volume of sets within the subspace $\{x : x_1 = x_2 = 0\}$ of $\mathbb{F}_2^{n+2}$ and its three cosets. Given a set $S \subseteq \mathbb{F}_2^{n+2}$ and a pair of bits $(b_1, b_2) \in \{0,1\}^2$, we define

$$\text{Vol}_{b_1 b_2}(S) := \frac{|S \cap \{x \in \mathbb{F}_2^{n+2} : x_1 = b_1, x_2 = b_2\}|}{2^n}.$$

in order to simplify notation.

▶ **Definition 15.** *Given $\varepsilon > 0$, we say that a set $S \subseteq \mathbb{F}_2^{n+2}$ is an* ε-eligible sumset *if $S = A + A$ for some $A \subseteq \mathbb{F}_2^{n+2}$ and if the following holds:*

$$\text{Vol}_{00}(S) \geq 1 - \varepsilon \qquad \textit{and} \qquad \text{Vol}_{11}(S) \leq \varepsilon.$$

Roughly, the ϵ-eligible sumsets are all those that might be close to $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$.

▶ **Proposition 16.** *For any ε, the number of ε-eligible sumsets in $\mathbb{F}_2^{n+2}$ is at most*

$$\max\left\{2^{4H(\epsilon)\cdot 2^n}, 2^{(1+2H(\epsilon))2^n}\right\} \cdot 2^{O(n)}.$$

We defer the proof of Proposition 16 to Appendix A. We conclude with the proof of Proposition 13.

Proof of Proposition 13. By Proposition 16, the number of ϵ-eligible sumsets is $2^{(1+2H(\epsilon))2^n} \cdot 2^{O(n)}$ when $\epsilon < 0.1$. By Equation (1), the number of *subsets* of $\mathbb{F}_2^{n+2}$ that are γ-close to a given sumset is

$$\binom{2^{n+2}}{\gamma 2^{n+2}} = 2^{H(\gamma)2^{n+2}} \cdot 2^{O(n)}.$$

Thus, by union-bounding over all ϵ-eligible sumsets, we conclude that the number of subsets of $\mathbb{F}_2^{n+2}$ that are γ-close to any ϵ-eligible sumset is at most

$$2^{(1+2H(\epsilon))2^n + H(\gamma)2^{n+2}} \cdot 2^{O(n)}.$$

Choosing $\epsilon = 0.05$ and $\gamma = \epsilon/4$ gives an upper bound of $2^{1.96\cdot 2^n}$ subsets of $\mathbb{F}_2^{n+2}$ that are $\epsilon/4$-close to any ϵ-eligible sumset. Since $\mathcal{S}_{\text{no}}$ is distributed uniformly over $2^{2^{n+1}}$ subsets, the probability that $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$ is $(\epsilon/4)$-close to any ϵ-eligible sumset is $2^{-\Omega(2^n)}$.

We further claim that $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$ is always $(\varepsilon/4)$-far from any sumset that is not ε-eligible. This is just because that we always have $\text{Vol}_{11}(\boldsymbol{S}_{\text{no}}) = 0$ and $\text{Vol}_{00}(\boldsymbol{S}_{\text{no}}) = 1$. On the other hand, any non-ϵ-eligible sumset S has either $\text{Vol}_{00}(S) < 1 - \epsilon$ or $\text{Vol}_{11}(S) > \epsilon$ by definition and thus, must be at least $(\epsilon/4)$-far from $\boldsymbol{S}_{\text{no}}$.

Thus with probability at least $1 - o_n(1)$, $\boldsymbol{S}_{\text{no}} \sim \mathcal{S}_{\text{no}}$ is $\epsilon/4 = 0.0125$-far from any sumset. ◀

6 Refuting Sumsets in the Smoothed Analysis Setting

In this section we study the smallest size of a *certificate* that a set is *not* a sumset. Informally, for a set $S \subseteq \mathbb{F}_2^n$ a sumset 0-certificate is a set $D \subseteq \mathbb{F}_2^n$ of points such that querying the oracle $\mathcal{O}_S$ on every point in D suffices to prove that S is not a sumset. More formally:

▶ **Definition 17.** *A set $D \subseteq \mathbb{F}_2^n$ is a* sumset 0-certificate for $S \subseteq \mathbb{F}_2^n$ *if there is no sumset $S' = A + A \subseteq \mathbb{F}_2^n$ for which $S \cap D = S' \cap D$.*

Small 0-certificates are important objects of study for many property testing problems; for example, consider the classic problem of linearity testing. Since a function $f : \mathbb{F}_2^n \to \mathbb{F}_2$ is linear if and only if $f(x + y) = f(x) + f(y)$ for all $x, y \in \mathbb{F}_2^n$, the property of linearity is characterized by the non-existence of a "linearity 0-certificate" of size three. As is well known, in the seminal work [12] Blum et al. showed that this is a *robust* characterization, in the sense that a simple sampling procedure which queries random triples $x, y, x + y$ and checks whether they constitute a linearity 0-certificate suffices to distinguish linear functions from functions which are far from being linear. A similar framework of sampling 0-certificates is at the heart of many other important property testing results such as low degree testing (see e.g. [5, 28] and many other works) and testing triangle-freeness (see e.g. [4, 2] and many other works). Of course, testing results of this sort rely on, and motivate the discovery of, structural results showing that functions which are far from having the property in question must have "many" "small" 0-certificates.

With this motivation, it is natural to study the size of sumset 0-certificates. Our sumset testing lower bound from Section 5 suggests that there are sets which are far from being sumsets but which do not have "many" "small" sumset 0-certificates. In fact, known results imply that for every non-sumset the smallest 0-certificate is of size $\Omega(2^{n/2}/\sqrt{n})$:

▶ **Lemma 18.** *Let $S \subseteq \mathbb{F}_2^n$ be any non-sumset. Then any sumset 0-certificate D for S must have $|D| \geq \Omega(2^{n/2}/\sqrt{n})$.*

Proof. This is an immediate corollary of a result due to Alon (Section 4 of [3]), which shows that any subset $T \subseteq \mathbb{F}_2^n$ of size $|T| \geq 2^n - \frac{1}{4000}\frac{2^{n/2}}{\sqrt{n}}$ is a sumset. It follows that if $|D| < \frac{1}{4000}\frac{2^{n/2}}{\sqrt{n}}$, then any 0/1 labeling of the points in D is consistent with a sumset (by labeling all points in $\mathbb{F}_2^n \setminus D$ as belonging to the set). ◀

The previous lemma, which establishes that any 0-certificate for sumset testing must have size $\Omega(2^{n/2}/\sqrt{n})$, establishes Part (2) of Theorem 3. In the full version of our paper, we prove a matching upper bound (up to a factor of $\mathrm{poly}(n, 1/\epsilon)$) for any set perturbed by a small amount of random noise, thereby establishing Part (1) of Theorem 3:

▶ **Theorem 19.** *For any set $S \subseteq \mathbb{F}_2^n$ and any $\varepsilon \in (0, \frac{1}{2}]$, there exists a sumset 0-certificate for $\boldsymbol{N}_\varepsilon(S) = S \triangle \boldsymbol{R}_\varepsilon$ of size $2^{n/2} \cdot O(n^{1.5}/\epsilon^{1.5})$, with probability $1 - o_n(1)$ over the random draw of $\boldsymbol{R}_\varepsilon$. Moreover, such a 0-certificate can be found efficiently and non-adaptively (by querying $2^{n/2} \cdot O(n^{1.5}/\epsilon^{1.5})$ points) given oracle access to $\boldsymbol{N}_\varepsilon(S)$.*

References

1 Amir Abboud, Nick Fischer, Ron Safier, and Nathan Wallheimer. Recognizing Sumsets is NP-Complete. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4484–4506, 2025. doi:10.1137/1.9781611978322.153.

2 N. Alon. Testing subgraphs in large graphs. *Random Structures Algorithms*, 21:359–370, 2002. doi:10.1002/RSA.10056.

3 N. Alon. Large sets in finite fields are sumsets. *Journal of Number Theory*, 126(1):110–118, 2007.

4 N. Alon, E. Fischer, M. Krivelevich, and M. Szegedy. Efficient testing of large graphs. *Combinatorica*, 20:451–476, 2000. doi:10.1007/S004930070001.

5 N. Alon, T. Kaufman, M. Krivelevich, S. Litsyn, and D. Ron. Testing low-degree polynomials over GF(2). In *Proc. RANDOM*, pages 188–199, 2003.

6 Noga Alon and Or Zamir. Sumsets in the hypercube. *SIAM Journal on Discrete Mathematics*, 39(1):314–326, 2025. doi:10.1137/24M165569X.

7 Boaz Barak, Russell Impagliazzo, and Avi Wigderson. Extracting randomness using few independent sources. *SIAM Journal on Computing*, 36(4):1095–1118, 2006. doi:10.1137/S0097539705447141.

8 Boaz Barak, Luca Trevisan, and Avi Wigderson. A mini-course on additive combinatorics, 2007. Available at https://www.math.cmu.edu/~af1p/Teaching/AdditiveCombinatorics/allnotes.pdf.

9 Eli Ben-Sasson, Shachar Lovett, and Noga Ron-Zewi. An Additive Combinatorics Approach Relating Rank to Communication Complexity. *J. ACM*, 61(4), July 2014. doi:10.1145/2629598.

10 Khodakhast Bibak. Additive combinatorics: With a view towards computer science and cryptography—an exposition. In Jonathan M. Borwein, Igor Shparlinski, and Wadim Zudilin, editors, *Number Theory and Related Fields*, pages 99–128, New York, NY, 2013. Springer New York. doi:10.1007/978-1-4614-6642-0_4.

11 Eric Blais. Testing juntas nearly optimally. In *Proc. 41st Annual ACM Symposium on Theory of Computing (STOC)*, pages 151–158, 2009. doi:10.1145/1536414.1536437.

12 M. Blum, M. Luby, and R. Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47:549–595, 1993. Earlier version in STOC'90. doi:10.1016/0022-0000(93)90044-W.

13 Jean Bourgain. More on the sum-product phenomenon in prime fields and its applications. *Internat. J. Number Theory*, 1(1):1–32, 2005. doi:10.1142/S1793042105000108.

14 Mark Braverman, Subhash Khot, and Dor Minzer. Parallel repetition for the GHZ game: Exponential decay. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 1337–1341. IEEE, 2023. doi:10.1109/FOCS57990.2023.00080.

15 Laurent Bulteau, Guillaume Fertin, Romeo Rizzi, and Stéphane Vialette. Some algorithmic results for [2]-sumset covers. *Information Processing Letters*, 115(1):1–5, 2015. doi:10.1016/J.IPL.2014.07.008.

16 Ashok K. Chandra, Merrick L. Furst, and Richard J. Lipton. Multi-party protocols. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, STOC '83, pages 94–99, New York, NY, USA, 1983. Association for Computing Machinery. doi:10.1145/800061.808737.

17 Michael J Collins, David Kempe, Jared Saia, and Maxwell Young. Nonnegative integral subset representations of integer sets. *Information Processing Letters*, 101(3):129–133, 2007. doi:10.1016/J.IPL.2006.08.007.

18 Ernie Croot and Seva Lev. Open problems in additive combinatorics. In *Additive Combinatorics*, volume 43 of *CRM Proceedings and Lecture Notes*, page 207. American Mathematical Society, 2007.

19 Anindya De, Shivam Nadimpalli, and Rocco A. Servedio. Approximating sumset size. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2339–2357. SIAM, 2022. doi:10.1137/1.9781611977073.94.

20 I. Diakonikolas, H. Lee, K. Matulef, K. Onak, R. Rubinfeld, R. Servedio, and A. Wan. Testing for concise representations. In *Proc. 48th Ann. Symposium on Computer Science (FOCS)*, pages 549–558, 2007.

21 Zeev Dvir and Amir Shpilka. An improved analysis of linear mergers. *Comput. Complex.*, 16(1):34–59, May 2007. doi:10.1007/s00037-007-0223-z.

22 Isabelle Fagnot, Guillaume Fertin, and Stéphane Vialette. On finding small 2-generating sets. In *Computing and Combinatorics: 15th Annual International Conference, COCOON 2009 Niagara Falls, NY, USA, July 13-15, 2009 Proceedings 15*, pages 378–387. Springer, 2009. doi:10.1007/978-3-642-02882-3_38.

23 G.A. Freiman. *Foundations of a Structural Theory of Set Addition.* Translations of mathematical monographs. American Mathematical Society, 1973. URL: https://books.google.com/books?id=8zc14FDkWlAC.

24 David Galvin. Independent sets in the discrete hypercube. arXiv preprint 1901.01991, 2019.

25 Chris Godsil and Brendan Rooney. Hardness of computing clique number and chromatic number for cayley graphs. *European Journal of Combinatorics*, 62:147–166, 2017. doi:10.1016/J.EJC.2016.12.005.

26 Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2):302–343, 2002. doi:10.1007/s00453-001-0078-7.

27 Ben J. Green. Finite field models in additive combinatorics. In Bridget S. Webb, editor, *Surveys in combinatorics*, pages 1–27. Cambridge Univ. Press, 2005.

28 Charanjit S. Jutla, Anindya C. Patthak, Atri Rudra, and David Zuckerman. Testing low-degree polynomials over prime fields. In *Proc. 45th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 423–432. IEEE Computer Society Press, 2004. doi:10.1109/FOCS.2004.64.

29 Subhash Khot, Dor Minzer, and Muli Safra. On monotonicity testing and boolean isoperimetric-type theorems. *SIAM Journal on Computing*, 47(6):2238–2276, 2018. doi:10.1137/16M1065872.

30 Shachar Lovett. *Additive Combinatorics and its Applications in Theoretical Computer Science.* Number 8 in Graduate Surveys. Theory of Computing Library, 2017. doi:10.4086/toc.gs.2017.008.

31 Anup Rao. An exposition of Bourgain's 2-source extractor. *Electronic Colloquium on Computational Complexity (ECCC)*, 14(34), 2007. ECCC.

32 Alex Samorodnitsky. Low-degree tests at large distances. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, STOC '07, pages 506–515, New York, NY, USA, 2007. Association for Computing Machinery. doi:10.1145/1250790.1250864.

33 V. G. Sargsyan. Counting Sumsets and Differences in an Abelian Group. *Journal of Applied and Industrial Mathematics*, 9(2):275–282, 2015.

34 Daniel A. Spielman and Shang-Hua Teng. Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time. In *Proceedings on 33rd Annual ACM Symposium on Theory of Computing*, pages 296–305, 2001. doi:10.1145/380752.380813.

35 Luca Trevisan. Additive combinatorics and theoretical computer science. *ACM SIGACT News*, 40(2):50–66, 2009.

36 Emanuele Viola. *Selected Results in Additive Combinatorics: An Exposition.* Number 3 in Graduate Surveys. Theory of Computing Library, 2011. doi:10.4086/toc.gs.2011.003.

A Proof of Proposition 16

Proof. Let S be any ϵ-eligible sumset, and let A satisfying $A + A = S$ be an additive root of S. We bound the number of ϵ-eligible sumsets by considering possibilities for A.

We begin with the observation that if $\mathrm{Vol}_{00}(A), \mathrm{Vol}_{11}(A) > 0$, then it must be true that $\mathrm{Vol}_{00}(A), \mathrm{Vol}_{11}(A) \le \epsilon$. Otherwise, we would have $\mathrm{Vol}_{11}(A+A) = \mathrm{Vol}_{11}(S) > \epsilon$, contradicting our assumption that S is ϵ-eligible. Likewise, we have that if $\mathrm{Vol}_{01}(A), \mathrm{Vol}_{10}(A) > 0$, then $\mathrm{Vol}_{01}(A), \mathrm{Vol}_{10}(A) \le \epsilon$. We split into cases accordingly.

1. All four cosets of $\{x : x_1 = x_2 = 0\}$ are nonempty:
   $\mathrm{Vol}_{00}(A), \mathrm{Vol}_{11}(A), \mathrm{Vol}_{01}(A), \mathrm{Vol}_{10}(A) > 0$.
   In this case, we have that $\mathrm{Vol}_{00}(A), \mathrm{Vol}_{11}(A)$, $\mathrm{Vol}_{01}(A), \mathrm{Vol}_{01}(A) \leq \epsilon$. Using Equation (1), we can then bound the number of possibilities for A (and S) by
   $$\binom{2^n}{\varepsilon 2^n}^4 \leq 2^{4H(\varepsilon)\cdot 2^n} \cdot 2^{O(n)}.$$
2. Either $\mathrm{Vol}_{00}(A)$ and $\mathrm{Vol}_{11}(A) > 0$, or $\mathrm{Vol}_{01}(A)$ and $\mathrm{Vol}_{10}(A) > 0$, but not both.
   Here the volume of A on two of the four cosets is at most ϵ, in one other coset it is 0, and in the final coset it may be as large as 1. In this case, again using Equation (1), the number of possibilities for A (and S) is bounded by
   $$2^{2^n} \cdot \binom{2^n}{\varepsilon 2^n}^2 \leq 2^{(1+2H(\varepsilon))\cdot 2^n} \cdot 2^{O(n)}.$$
3. Either $\mathrm{Vol}_{00}(A)$ or $\mathrm{Vol}_{11}(A) = 0$, and either $\mathrm{Vol}_{01}(A)$ or $\mathrm{Vol}_{10}(A) = 0$, hence at least two of the four cosets contain no points in A.
   Assume without loss of generality that $\mathrm{Vol}_{11}(A) = \mathrm{Vol}_{01}(A) = 0$. This immediately implies that $\mathrm{Vol}_{01}(S) = \mathrm{Vol}_{11}(S) = 0$, as $A + A$ cannot contain points in either coset. (Note that, whichever pair of cosets we choose to zero out, this implies that $\mathrm{Vol}_{11}(S) = 0$ and either that $\mathrm{Vol}_{01}(S) = 0$ or $\mathrm{Vol}_{10}(S) = 0$.) Using the fact that $\mathrm{Vol}_{00}(S) \geq 1 - \epsilon$, the number of possibilities for S is bounded by
   $$2^{2^n} \cdot \binom{2^n}{\varepsilon 2^n} \leq 2^{(1+H(\varepsilon))\cdot 2^n} \cdot 2^{O(n)}.$$

Summing the number of ϵ-eligible sumsets covered by each case completes the proof. ◀