Homomorphism Testing with Resilience to Online Manipulations

Esty Kelman
Massachusetts Institute of Technology, Cambridge, MA, USA
Boston University, MA, USA

Uri Meir
Blavatnik School of Computer Science, Tel Aviv University, Israel

Debanuj Nayak
Boston University, MA, USA

Sofya Raskhodnikova
Boston University, MA, USA

Abstract

A central challenge in property testing is verifying algebraic structure with minimal access to data. A landmark result addressing this challenge, the linearity test of Blum, Luby, and Rubinfeld (JCSS '93), spurred a rich body of work on testing algebraic properties such as linearity and its generalizations to low-degree polynomials and group homomorphisms. However, classical tests for these properties assume unrestricted, noise-free access to the input function – an assumption that breaks down in adversarial or dynamic settings. To address this, Kalemaj, Raskhodnikova, and Varma (Theory of Computing '23) introduced the online manipulation model, where an adversary erases or corrupts query responses over time, based on the tester's past queries.

We initiate the study of *manipulation-resilient* testing for *group homomorphism* in this online model. Our main result is an *optimal* tester that makes $O(1/\varepsilon + \log t)$ queries, where ε is the distance parameter and t is the number of function values the adversary can erase or corrupt per query. Our result recovers the celebrated $O(1/\varepsilon)$ bound by Ben-Or, Coppersmith, Luby, and Rubinfeld (Random Struct. Algorithms '08) for homomorphism testing in the standard property testing model, albeit with a different tester. Our tester, Random Signs Test, *lifts* known manipulation-resilient linearity testers for $\mathbb{F}_2^n \to \mathbb{F}_2$ to general group domains and codomains by introducing more randomness: instead of verifying the homomorphism condition for a sum of random elements, it uses additions and subtractions of random elements, randomly selecting a sign for each element. We also obtain improved group-specific query bounds for key families of groups. Our results show that despite the challenges of online manipulation, group homomorphism – a fundamental algebraic property – is efficiently testable across a wide range of domains and codomains. Finally, we formalize a general framework for proving online resilience.

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

Keywords and phrases Property Testing, Sublinear Algorithms, Online Manipulation Resilience, Group Theory

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2026.58

Category RANDOM

Related Version *Full Version*: `https://arxiv.org/abs/2511.23363` [35]

Funding *Esty Kelman*: Partially supported by NSF awards DMS-2022448, DMS-2022446, and CCF-2227876.
Debanuj Nayak: Supported in part by NSF award 2022446.
Sofya Raskhodnikova: Supported in part by NSF award 2022446.

Acknowledgements The authors would like to thank Ephraim Linder for useful discussions. Part of this research was carried out during the authors' visit to the Simons Institute for the Theory of Computing at UCB.

1 Introduction

Understanding how to verify algebraic structure with minimal access to data is a central question in *property testing*. A landmark result addressing this challenge is the linearity test of Blum, Luby, and Rubinfeld [15], which checks whether a given function is linear by querying it on two random inputs and their sum. This test and its variants sparked a rich line of work on testing algebraic properties of functions, including linearity [6, 21, 8], being a low-degree polynomial [5, 23, 22, 53, 51, 1, 3, 46, 33, 54, 56, 30, 14, 28, 52, 17, 32], and being a group homomorphism [15, 27, 11, 25, 7, 45, 26, 43], with deep connections to probabilistically-checkable proofs [2, 59, 29].

However, classical property testing, as defined by [53, 24], relies on the assumption of undisrupted query access to the input function – an assumption that may break down in dynamic environments or under adversarial or incomplete access to data. Many testers, especially for algebraic properties, use structured queries that can be vulnerable to online adversarial interference. To address this, Kalemaj, Raskhodnikova, and Varma [31] introduced the *online manipulation model,* where an adversary can erase or corrupt the input during the execution of the algorithm, based on prior queries. This framework has prompted the reexamination of classical testing problems under a new lens, seeking algorithms that remain correct and efficient despite adversarial interference.

In this work, we initiate the study of manipulation-resilient testing for group homomorphism, a fundamental property of functions between algebraic structures. Given finite groups $(G, +)$ and $(H, \oplus)$, and a function $f : G \to H$, the goal is to determine whether f is a *homomorphism*[1] – i.e., satisfies $f(x_1 + x_2) = f(x_1) \oplus f(x_2)$ for all $x_1, x_2 \in G$ – or is ε-far from every homomorphism.[2] The tester accesses the input function f by querying its values $f(x)$ on elements x of the group, but must contend with an *online adversary* that knows all queries made by the tester and can manipulate the data after answering each query. Two types of manipulation are considered: an *erasure*, where $f(x)$ is replaced by a special symbol $\perp$, and a *corruption*, where $f(x)$ is replaced by an arbitrary element of H. The rate of manipulations is controlled by a parameter t. Two types of adversaries have been studied: *fixed rate*, which can manipulate t values after every query is answered [31], and *budget-managing*, which accumulates a budget of t manipulations per query and may deploy them later at any point in the computation, after answering any query [10].

Linearity and low-degree testing are well understood in the online adversarial model. Linearity and quadraticity were among the first properties studied in this setting, with [31] proving a query lower bound of $\Omega(\log t)$ for testing linearity of functions $f : \mathbb{F}_2^n \to \mathbb{F}_2$. fThe linearity tester of [31] was improved by Ben-Eliezer, Kelman, Meir and Raskhodnikova [10], who achieved the optimal query complexity of $O(1/\varepsilon + \log t)$, and further refined by Arora, Kelman, and Meir [4] to handle the full range of the parameter t. For the related problem of direct sum testing, Westover, Yu and Zheng [60] achieve the same query complexity. For testing whether a function $f : \mathbb{F}_q^n \to \mathbb{F}_q$ has degree at most d, for fixed constants d, q and ε, a lower bound of $\Omega(\log^d t)$ on the query complexity was first shown by [10] for $q = 2$ and later by [36] whenever $q \geq d$, whereas the elegant tester of Minzer and Zheng [42] makes $\text{poly}(\log^d t)$ queries, and a recent refined tester and analysis yield the tight upper bound of $O(\log^d t)$ queries for this parameter regime [36].

[1] As noted in the Encyclopedia of Algorithms [50], the set of homomorphisms from G to H forms an error-correcting code, called the *homomorphism code*. When $G = \mathbb{F}_2^n$ and $H = \mathbb{F}_2$, this corresponds to the well-known Hadamard code. From this perspective, homomorphism testing is equivalent to testing membership in the corresponding code.

[2] We use additive notation for all group operations, even for non-abelian groups (as opposed to using $\cdot$ and $*$), as it enables compact summation notation (specifically, $\sum$ and $\bigoplus$) and leads to more concise and readable expressions.

While both lower bounds [31, 10] apply to the weakest model of online adversary (fixed-rate with erasures), the algorithms of [10, 4, 42, 60] are resilient even in the strongest model considered (budget-managing with corruptions).

These developments raise a natural question: Can one design testers resilient to online manipulations for the other natural generalization of linearity – namely, group homomorphism?[3] We answer this question affirmatively by designing testers that are resilient to the strongest type of online adversary in our framework – budget-managing with corruptions. Our main result is that homomorphism of functions $f : G \to H$ can be tested with $O(1/\varepsilon + \log t)$ queries in the presence of a t-online corruption adversary for all groups G and H. This bound is optimal: even without manipulations, $\Omega(1/\varepsilon)$ queries are required, and the $\Omega(\log t)$ lower bound of [31] holds for the special case $G = \mathbb{F}_2^n$ and $H = \mathbb{F}_2$.

Since the lower bound applies to specific groups G and H, a natural next question is whether some groups admit better testers. We obtain improved group-specific query bounds for many important families of groups. In particular, we design an online-manipulation-resilient homomorphism tester that makes $O(1/\varepsilon + E(G))$ queries, where $E(G)$ is the expected number of samples from G needed to generate G. This tester is optimal, for instance, when G is a cyclic, symmetric, alternating, or simple group. For vector spaces over finite fields of prime order, we obtain nearly tight bounds: when $G = \mathbb{F}_p^n$ and $H = \mathbb{F}_p^r$, homomorphism can be tested with $O(1/\varepsilon + \log_p t)$ queries, and we prove a matching $\Omega(\log_p t)$ lower bound for the case $r = 1$; when $H = \mathbb{F}_q^r$ where $q \neq p$, testing reduces to checking whether f is identically zero.

Our results demonstrate that group homomorphism – a fundamental algebraic property – admits efficient and manipulation-resilient testers across the board: we give a general optimal tester for all finite groups and complement it with sharper bounds for prominent families of domains and codomains.

1.1 Our results

Our main result is an *optimal* online-manipulations-resilient homomorphism tester of $f : G \to H$ for all groups G and H. As all our testers, it works in the strongest online adversary model considered: budget-managing with corruptions.

▶ **Theorem 1.1.** *There exists a constant $c > 0$ such that for all finite groups $(G, +)$ and $(H, \oplus)$, all $\varepsilon \in (0, 1)$ and $t \leq c \cdot \min\left\{\varepsilon^2, 1/\log^2 |G|\right\} \cdot |G|$, there exists an ε-tester for group homomorphism of functions of the form $f : G \to H$ that works in the presence of every budget-managing t-online erasure (or corruption) adversary and makes $O\left(\frac{1}{\varepsilon} + \log_2 t\right)$ queries. For erasures, the tester has 1-sided error.*

Our result recovers the celebrated $O(1/\varepsilon)$ bound by [15, 11] for testing homomorphism in the standard property testing model, albeit with a different tester. Our tester, which we call Random Signs Test, can be viewed as *lifting* the linearity testers developed by [31, 10, 4] for functions $f : \mathbb{F}_2^n \to \mathbb{F}_2$ to work for general groups G and H. As noted by [31], the original [15] tester breaks in the online manipulations model, and as we explain in Section 1.2, the known online-manipulation-resilient testers for the case $G = \mathbb{F}_2^n$ and $H = \mathbb{F}_2$ are not sound for general groups. Let $(G, +)$ be the domain group and $(H, \oplus)$ be the codomain group. For

[3] Importantly, efficient testability of a property $\mathcal{P}$ in the standard property testing model does not guarantee that $\mathcal{P}$ is testable with online adversary. For instance, [31] show that *sortedness* of integer sequences – despite having many efficient testers in the standard model [20, 19, 48, 13, 16, 49, 9] – is not testable even with a single online erasure per query, regardless of the number of queries.

the case without manipulations, our tester picks k points (for any even k) $x_1, \ldots, x_k$ from the domain group G and k signs $\sigma_1, \ldots, \sigma_k$ from $\{+, -\}^k$ uniformly and independently at random. Then it obtains an element $a = \sigma_1 x_1 + \cdots + \sigma_k x_k$, where $\sigma_i x_i$ denotes x_i when $\sigma_i = +$ and the inverse of x_i (in G) when $\sigma_i = -$. Finally, it queries f on $x_1, \ldots, x_k$, and a, and tests whether the homomorphism condition is satisfied for these elements, that is, whether $f(a) = \sigma_1 f(x_1) \oplus \cdots \oplus \sigma_k f(x_k)$, where $\sigma_i = -$ denotes the inverse in the group H. The manipulation-resilient version of the tester first samples a reserve of elements of G and then runs Random Signs Test with carefully selected k and elements $x_1, \ldots, x_k$ from the reserve. We provide a technical overview of the tester and its analysis in Section 1.2. Section 3 presents and analyzes Random Signs Test, and Section 5 – a manipulation-resilient version of Random Signs Test.

1.1.1 Group-specific homomorphism testing

Our first group-specific result, Theorem 1.3, gives a homomorphism tester with complexity expressed in terms of the parameter $E(G)$, defined next.

▶ **Definition 1.2** ([47]). *For all finite groups $(G, +)$, let $E(G)$ be the expected number of independent and uniform samples from G needed to obtain a generating set for G.*

This group-specific tester has an additional advantage of being *sample-based*, i.e., querying only independent and uniform elements of G. Its guarantees are given in Theorem 1.3 and proved in the full version.

▶ **Theorem 1.3** (Group-specific sample-based tester). *There exists a constant $c > 0$ such that for all finite groups $(G, +)$ and $(H, \oplus)$, all $\varepsilon \in (0, 1)$ and $t \leq c \cdot \min\left\{\varepsilon^2, 1/E(G)^2\right\} \cdot |G|$, there exists a sample-based ε-tester for group homomorphism of functions of the form $f : G \to H$ that works in the presence of every t-online erasure (or corruption) budget-managing adversary and makes $O(1/\varepsilon + E(G))$ queries. For erasures, the tester has 1-sided error.*

In the model without manipulations, Goldreich and Ron [25] gave a sample-based homomorphism tester with $O(1/\varepsilon + \log |G|)$ queries and showed this bound is tight for some groups G and H (in particular, $G = \mathbb{F}_2^n, H = \mathbb{F}_2$). The query bound in Theorem 1.3 improves on the bound in [25] for many natural groups and is tight when $E(G)$ is constant – e.g., for cyclic groups of prime order $\mathbb{Z}_p$, symmetric groups S_n and alternating groups A_n [18], and, more generally, finite simple groups [38]. This gives us the following corollary.

▶ **Corollary 1.4** (Homomorphism tester for simple groups). *There exists a constant $c > 0$ such that for all finite groups $(G, +)$ and $(H, \oplus)$, where G is a finite simple group, all $\varepsilon \in (0, 1)$ and all $t \leq c \cdot \varepsilon^2 |G|$, there exists a sample-based ε-tester for group homomorphism of functions of the form $f : G \to H$ that works in the presence of a t-online-erasure adversary and makes $\Theta(1/\varepsilon)$ queries.*

To improve the bound in [25] and prove Theorem 1.3, we analyze the subgroup generated by the sample S, rather than only partial sums that can be obtained from S, obtaining better group-specific bounds on the number of samples needed to properly learn the input homomorphism. Our analysis relies on work in group theory [18, 38, 47, 39, 40, 41] that relates $E(G)$ from Definition 1.2 to the size of the smallest set generating G and other structural parameters of the group.

Bounds for prime fields. Finally, we consider groups of the form $G = \mathbb{F}_p^n$ and $H = \mathbb{F}_q^r$, where p and q are primes and $n, r \in \mathbb{N}$. Our next result shows that if $p = q$, the query complexity is below the general bound of $O(1/\varepsilon + \log_2 t)$ from Theorem 1.1.

▶ **Theorem 1.5.** *There exists a constant $c > 0$ such that for all primes p, all $n, r \in \mathbb{N}$, all $\varepsilon \in (0,1)$ and $t \leq c \cdot \min\left\{\varepsilon^2, 1/n^2\right\} p^n$, there exists an ε-tester for homomorphism of functions of the form $f : \mathbb{F}_p^n \to \mathbb{F}_p^r$ that works in the presence of every t-online erasure (or corruption) budget-managing adversary and makes $O(1/\varepsilon + \log_p t)$ queries. For erasures, the tester has 1-sided error.*

Next we show that, when the range is $\mathbb{F}_p$ ($r = 1$), the bound in Theorem 1.5 is tight.

▶ **Theorem 1.6.** *For all primes p, there exists $n_0 \in \mathbb{N}$ such that for all $n \in \mathbb{N}, n \geq n_0$ and $\varepsilon \in (0, \frac{p-1}{2p}]$, every t-online erasure-resilient ε-tester for homomorphism of functions of the form $f : \mathbb{F}_p^n \to \mathbb{F}_p$ must make $\Omega(\log_p t)$ queries.*

We defer the proofs of Theorems 1.5 and 1.6 to the full version. Finally, we observe that the case when $p \neq q$ is easy.

▶ Remark 1.7. For all finite groups G and H such that the zero map is the only homomorphism from G to H (e.g., when $G = \mathbb{F}_p^n$ and $H = \mathbb{F}_q^r$, where p and q are distinct primes and $n, r \in \mathbb{N}$), testing if a function $f : G \to H$ is a homomorphism is equivalent to testing if f is identically zero and can be done with $\Theta(1/\varepsilon)$ queries, even in the presence of a t-online manipulation budget-managing adversary.

1.1.2 A general framework for resilience

Resilience to online manipulations creates a fundamental tension: a tester must make queries that are hard for the adversary to predict, yet these may not be the most informative ones. This perspective naturally interpolates between two familiar extremes in property testing: query-based testers, which optimize for information useful for the specified computational task but may be predictable, and sample-based testers, which query f on independent uniformly random elements of G and thus maximize unpredictability. Understanding how to balance these competing goals provides a new lens on property testing.

To formalize this perspective, we need a notion of unpredictability that captures this trade-off. A natural candidate is the notion of *fairness* of Goldreich and Ron [25], which constrains the marginal distribution of each query. A tester is *α-fair* if, at every step, no $x \in G$ is queried with probability more than α. They showed that if a property has an α-fair ε-tester that makes q queries with $\alpha = \Theta(|G|^{-1})$ and constant ε then it also has a sample-based tester (in a sense, a "completely fair" tester) with $|G|^{1-\frac{1}{q}}$ queries. This provides an intriguing trade-off between the number of queries and their predictability. However, fairness is ill-suited for resilience to online manipulations, as it does not account for dependencies between queries. For example, the [15] linearity test is completely fair – each query is uniformly random – yet, as observed by [31], it is vulnerable to online manipulations because its third query is fully determined by the first two. This suggests that marginal notions of unpredictability are insufficient.

The key ingredient in our framework for proving resilience is a new notion of *conditional* unpredictability for testers (Definition 4.2). Informally, a tester is (q, α, β)-unpredictable if, with probability at least $1 - \beta$, *each* of its q queries remains hard to predict even after conditioning on the previous queries. Formally, there exist parameters $\{\alpha_i, \beta_i\}_{i \in [q]}$ with $\sum_{i \in [q]} \alpha_i = \alpha$ and $\sum_{i \in [q]} \beta_i = \beta$ such that for each $i \in [q]$, with probability at least $1 - \beta_i$, the

conditional distribution of the ith query Q_i is α_i-flat, i.e., $\max_{x \in G} \Pr[Q_i = x \mid Q_1, \ldots, Q_{i-1}] \leq \alpha_i$. This sequential, conditional structure is reminiscent of Santha–Vazirani sources from the extractor literature [57].

We show that unpredictability directly yields resilience to online manipulations: regardless of the past manipulations, an unpredictable tester is unlikely to ever query a manipulated entry. In particular, a (q, α, β)-unpredictable tester queries a manipulated location with probability at most $\alpha T + \beta$, where T is the total number of manipulations over the entire execution (see Lemma 4.3). To prove our main results, we first establish soundness of a *core test*, modify it to get an *unpredictable core test* which has the same soundness as the core test along with sufficient unpredictability. To get our final testers, we then amplify the success probability of the unpredictable core test by repetition. We address some nuances of amplification in the online manipulation model – where accumulated manipulations can make later repetitions less reliable – and prove an amplification lemma that specifies the required parameters of the unpredictable core test and the number of repetitions.

1.2 Technical overview of the tester for general groups

The starting point for the classical homomorphism testers, as well as for manipulation-resilient linearity testers, is a core test (equivalently, a local characterization of a property). All core tests we discuss have perfect *completeness:* they always accept homomorphisms. In addition, they need to have good *soundness:* high enough rejection probability of inputs that are far from the desired property. For the standard model, good soundness suffices, since repeating the core test amplifies its rejection probability and yields the desired tester. For the online adversarial model, however, soundness alone is insufficient; the final tester must also be unpredictable enough to prevent the adversary from targeting its queries. As in [31], this is achieved by first building a reservoir of uniformly random queried elements, then using a final query to simulate the core test on a random subset of the reservoir, and finally repeating the resulting procedure as needed.

Our main technical contribution is the design and analysis of a new core test. While its full analysis with the reservoir requires additional ideas, we focus here on why a natural extension of the manipulation-resilient linearity test fails in our setting, how our test is constructed, and the key insights behind its correctness.

Before introducing our tester, we discuss why a natural generalization of online manipulation resilient testers from previous work fails to provide soundness. The $(k+1)$-point linearity test of [31] queries the input function f on k independent and uniformly random elements $x_1, \ldots, x_k$ and their sum $x_{k+1} := \sum_{i \in [k]} x_i$ and checks whether $\sum_{i \in [k]} f(x_i)$ equals $f(\sum_{i \in [k]} x_i)$. Let $(G, +)$ and $(H, \oplus)$ be finite groups and $f : G \to H$ be a function. A natural extension of the [31] test to the homomorphism property is to check whether[4]

$$f(x_1) \oplus f(x_2) \oplus \cdots \oplus f(x_k) \stackrel{?}{=} f(x_1 + x_2 + \cdots + x_k). \tag{1}$$

If f is a homomorphism, then (1) holds for all tuples $(x_1, \ldots, x_k)$, but the converse is false: for $G = H = \mathbb{Z}_{k-1}$ (the additive cyclic group over $k-1$ elements) and any homomorphism $h : G \to H$, the shifted homomorphism function $f(x) = h(x) \oplus s$ with shift $s \in H$ also satisfies (1) for all tuples.[5] To resolve this issue, our homomorphism tester uses a more robust local characterization.

---

[4] We use compact notation to represent our sums: (1) can also be written as $\bigoplus_{i \in [k]} f(x_i) = f\left(\sum_{i \in [k]} x_i\right)$. In non-Abelian groups, the order matters. The terms in the sums are ordered according to the index: in increasing index order for $\sum_{i \in [k]} x_i$ or $\bigoplus_{i=1}^{k} y_i$, and in decreasing index order for $\sum_{i=k}^{1} x_i$.

[5] A similar phenomenon occurs for functions $f : \mathbb{F}_2^n \to \mathbb{F}_2$: the k-point tester with even k works for linearity, but odd k produces a test for *affinity* instead.

Next, we discuss our desired soundness guarantee and sketch a simple argument showing that many NO instances are rejected by the $(k+1)$-point test with sufficient probability. (The example above shows that this tester fails on functions that are really far from any homomorphism, but it turns out it works well for functions which are closer to a homomorphism). Let h_f be a closest homomorphism to f and ε_f be the distance from f to h_f. Define $\Delta_f := \{x \in G : f(x) \neq h_f(x)\}$ (so that $|\Delta_f| = \varepsilon_f \cdot |G|$). Since all tuples $(x_1, \ldots, x_k)$ satisfy modified (1) with f replaced by h_f, one could analyze for how many tuples (1) holds for f by comparing f and h_f on each tuple. The following "wishful thinking" argument is an easy assertion when ε_f is small enough, and was used before, e.g., for optimal analysis of low-degree testing over $\mathbb{F}_2$ [14].

Simple case: $\varepsilon_f < \frac{1}{2k}$. We view each point x_i (including the sum x_{k+1}) as a random variable, and ask whether $x_i \in \Delta_f$. Marginally, each x_i is distributed uniformly over G and so this occurs with probability ε_f for all $i \in [k+1]$. Intuitively, the points are "independent enough" so when $\varepsilon_f < \frac{1}{2k}$, w.h.p. at most one of them falls in Δ_f. If exactly one point does, then the tester rejects f (by cancellation rule on the group H). Formally, when $\varepsilon_f < \frac{1}{2k}$ the probability of exactly one x_i falling in Δ_f can be lower bounded by $k\varepsilon_f/2$ using the premise, pairwise independence of the points and inclusion-exclusion (Bonferroni inequality of order 2), as was done, e.g., in [14, 10] to analyze low-degree tests of polynomials over fields. By a union bound, the rejection probability is at most $k\varepsilon_f$ so the analysis above is optimal (up to a factor of 2) for inputs with small distance ε_f.

However, larger values of ε_f lead to more than a single point in Δ_f for a typical tuple, and the guaranteed pairwise independence is no longer enough. Luckily, the $k+1$ points used in (1) are in fact k-wise independent, which is a stronger guarantee for $k > 2$. We demonstrate how to leverage this fact, focusing on the case $H = \mathbb{F}_2$.

The elementary argument (for $\varepsilon_f < \frac{1}{8}$). Let p_m be the probability that m independent and uniformly random elements $z_1, \ldots, z_m \in G$ satisfy $\bigoplus_{i\in[m]} f(z_i) = \bigoplus_{i\in[m]} h_f(z_i)$. Let $\gamma_m = 1 - p_m$ be the probability of an inequality. By definition, $\gamma_1 = \varepsilon_f$. We utilize the k-wise independence by splitting the last point, $x_{k+1} = \sum_{i\in[k]} x_i$, from the rest and comparing the RHS and LHS of (1) with f to those with h_f where the equality holds. In particular, the event that the RHS of f and h_f agree but the LHS are different, leads to rejection of f. Then

$$\begin{aligned}
&\Pr_{x_1,\ldots,x_k\in G}\left[\bigoplus_{i\in[k]} f(x_i) \neq f\Big(\sum_{i\in[k]} x_i\Big)\right] \\
\geq &\Pr_{x_1,\ldots,x_k\in G}\left[\bigoplus_{i\in[k]} f(x_i) \neq \bigoplus_{i\in[k]} h_f(x_i) \ \wedge\ f\Big(\sum_{i\in[k]} x_i\Big) = h_f\Big(\sum_{i\in[k]} x_i\Big)\right] \\
\geq &\Pr\left[\bigoplus_{i\in[k]} f(x_i) \neq \bigoplus_{i\in[k]} h_f(x_i)\right] - \Pr\left[f\Big(\sum_{i\in[k]} x_i\Big) \neq h_f\Big(\sum_{i\in[k]} x_i\Big)\right] = \gamma_k - \gamma_1,
\end{aligned}$$

where the second inequality uses $\Pr[A \wedge B] \geq \Pr[A] - \Pr[\bar{B}]$ for two events A, B (where $\bar{B}$ is the complement of B). The equality is since $(x_1, \ldots, x_k)$ are i.i.d. samples taken uniformly at random from G.

From here on, the goal is to lower bound γ_k. To showcase the argument we restrict ourselves to $H = \mathbb{F}_2$ (which in particular captures linearity over $\mathbb{F}_2$), and discuss the extension later. Here, $f(x) - h_f(x) = 1$ if $x \in \Delta_f$, and otherwise $f(x) = h_f(x)$. That is, γ_k is the probability that there is an odd number of indices $i \in [k]$ for which $x_i \in \Delta_f$. Since these points are independent and each has probability ε_f to fall inside Δ_f, this equals to the

probability of a binomial random variable $Z \sim Bin(k, \varepsilon_f)$ being odd, which is known to equal $\frac{1-(1-2\varepsilon_f)^k}{2}$. This expression has a known lower bound of $\Omega(\min\{k\varepsilon_f, 1\})$, which absorbs the subtraction of $\gamma_1 = \varepsilon_f$ for any $\varepsilon_f \leq 1/8$ and completes the proof.[6]

To generalize this argument, note that $H = \mathbb{F}_2$ is essentially the worst-case scenario, as "coincidental cancellations" are most probable when $f(x) - h_f(x) \in \{0, 1\}$. In Section 3.1 we prove the claim for our modified tester, described shortly, and any codomain H, by analyzing an inductive recurrence directly on p_m (rather than $\gamma_m = 1 - p_m$).

The random signs test. We make the local characterization more robust by adding *signs*. We rely on the inverse operation, which exists in every group, and use $-x$ to express the addition of the inverse of x for both groups, G and H. Since every homomorphism satisfies $f(-x) = -f(x)$ for all $x \in G$, it respects not only summations of elements but signed summations as well. The corresponding tester is this: given a function $f : G \to H$, choose uniformly at random (and independently) a tuple $(x_1, \ldots, x_k, \sigma_1, \ldots, \sigma_k)$ consisting of k elements $x_1, \ldots, x_k \in G$, and k signs $\sigma_1, \ldots, \sigma_k \in \{+, -\}$. Finally, check whether

$$\bigoplus_{i\in[k]} \sigma_i f(x_i) \stackrel{?}{=} f(\sum_{i\in[k]} \sigma_i x_i). \tag{2}$$

As before, if f is a homomorphism, all tuples satisfy equation (2). The difference is that now, when k is even, the converse holds as well.[7] To analyze the soundness of Random Signs Test, we consider two cases. For $\varepsilon_f < 1/8$, the elementary argument (Lemma 3.2) suffices. To prove this tester rejects functions with $\varepsilon_f \geq 1/8$ (unlike the unmodified k-point tester), we show the contrapositive via a corrector argument based on that of [15, 11].

In the corrector method, a corrector function $g(x)$ is defined as the value satisfying the most instances of equation (2), where each instance corresponds to a signed tuple for which $\sum_{i\in[k]} \sigma_i x_i = x$. Differently put, each tuple of elements and signs that sum to x "votes" for a value it wishes $f(x)$ to be – then $g(x)$ is defined as the plurality of these votes. To prove soundness, one shows that if the test rejects f with probability $\mu \leq 1/10$, then two key assertions follow: (a) f is 2μ-close to g; (b) g is a homomorphism. The two assertions are proved in Lemmas 3.12 and 3.13, respectively, with the latter employing the probabilistic method. Both rely on an auxiliary step, Lemma 3.11, which shows that the choice of $g(a)$ as a plurality of votes, is in fact a strong majority of at least $1 - 2\mu$ of the votes, for every input $a \in G$.

We focus here on the proof of this step, which demonstrates the techniques of the full argument, and the role of random signs. The proof analyzes the collision probability – the probability that two signed tuples $(\sigma, x), (\tau, y)$ that sum to a vote for the same value for $f(a)$.

---

[6] We use $\frac{1-(1-2\varepsilon_f)^k}{2} = \Omega(\min\{k\varepsilon_f, 1\})$ with a specific constant, by applying [10, Lemma 3.1].

[7] Indeed, if f always satisfies equation (2), then given $x_1, x_2 \in G$ we use the tuple with elements x_1, x_2 and $x_j = 0$ for $j \geq 3$, and signs $\sigma_1 = \sigma_2 = ' + '$, and σ_j for $j \geq 3$ alternates between addition and subtraction. The $k - 2$ alterations of $+0, -0$ cancel out on the RHS, and similarly the alterations of $+f(0) - f(0)$ cancel out on the LHS (even if $f(0) \neq 0_H$), which asserts that $f(x_1) + f(x_2) = f(x_1 + x_2)$. This holds for any $x_1, x_2 \in G$, showing that f is a homomorphism.

By re-arranging, we get

$$\underbrace{\bigoplus_{i\in[2k]} \sigma_i f(x_i)}_{S_{\sigma,x}} = \underbrace{\bigoplus_{i\in[2k]} \tau_i f(y_i)}_{S_{\tau,y}} \iff$$

$$\underbrace{\bigoplus_{i=0}^{k-1} -\tau_{k-i} f(y_{k-i}) \oplus \bigoplus_{i\in[k]} \sigma_i f(x_i)}_{S_1} = \underbrace{\bigoplus_{i=k+1}^{2k} \tau_i f(y_i) \oplus \bigoplus_{i=0}^{k-1} -\sigma_{2k-i} f(x_{2k-i})}_{S_2}.$$

Crucially, the tuples underlying the sums S_1, S_2 are each a mix from (σ, x) and (τ, y) and we show that both have the distribution of the random sign test, but they sum (in G) to the same element, a'. Therefore $\Pr[S_{\sigma,x} \neq S_{\tau,y}] = \Pr[S_1 \neq S_2] \leq \Pr[S_1 \neq f(a') \vee S_2 \neq f(a')] \leq \Pr[S_1 \neq f(a')] + \Pr[S_2 \neq f(a')] = 2\mu$.

The necessity of random signs is apparent: without them, the sums $S_{\sigma,x}, S_{\tau,y}, S_1, S_2$ correspond to tuples with different sign vector, failing to relate the probability of this event to μ (the rejection probability of the test). By adding random signs, each element has a symmetric role in the test, which leaves the connection between test and equation even after one rearranges the elements, which is a delicate matter especially in non-abelian groups. Similar issues arise for odd k, as there is no way to split the two summations "down the middle". Both technical requirements appear to be tight: we have seen a choice for G, H where fixed signs (say, all are $+$) fail to characterize homomorphisms. Yet for $G = \mathbb{F}_2^n, H = \mathbb{F}_2$ the signs have no meaning ($x = -x$ always), and our test coincides with the simple k-point test, which requires an even value of k to test linearity (otherwise, it tests affinity).

1.3 Additional related work

Injecting more randomness vs. derandomization. A significant effort [59, 55, 29, 12, 58] went into reducing the number of random bits used by homomorphism tests. Interestingly, the online manipulation model presents a contrasting goal: increasing randomness to make future queries less predictable to the adversary. Indeed, Kelman, Linder and Raskhodnikova [34] show that for certain properties, testing in the online model requires exponentially more random bits than in the offline setting (even with erasures), despite comparable query complexity. This increased randomness requirement motivates the design of testers that are more *sample-based*, i.e., use uniformly random labeled samples in place of some arbitrary (potentially adaptive) queries. This trend aligns with learning-theoretic settings, where labeled samples are typically modeled as less costly than arbitrary queries.

Low-soundness regime. Our work focuses on the *high-soundness* regime (which relates to unique decoding), showing that if the input function f is accepted by the tester with high probability, then f is close to a homomorphism. In contrast, in the *low-soundness* regime, the goal is to show that small, yet non-trivial, acceptance probability by the tester implies the input function has a non-trivial structure resembling that of a homomorphism. Homomorphism testing in the low-soundness regime has been extensively studied. Only Khot and Mittal [37] use a k-point tester and their work is specific to linearity over the *biased* hypercube. Their tester chooses points according to Hamming weight rather than uniformly; it is unclear whether it is resilient to online manipulations, since the adversary can manipulate points that are more likely to be queried.

Concurrent work. Concurrently with our work, Mittal and Roy [44] presented a general framework for homomorphism testing in the low-soundness regime and applied it to various choices of domain group G and codomain group H. Specifically, for the case of vector spaces over prime fields, [44] presented a k-point test with random coefficients, identical to the test we design for the special case of such G, H (which appears on the full version). Curiously, the same algorithm is well-suited for both settings (for this specific choice of G and H): the low-soundness regime in the standard model and high soundness in the online manipulations model. The general version of their k-point test chooses a k-tuple from G according to a weight that, roughly, comes from the order of each of its elements in the group G. In contrast, our tester picks k-tuples uniformly at random for any choice of G and H to evade adversarial manipulations.

1.4 Open questions

Our main results are in a sense orthogonal: Theorem 1.1 applies to all finite groups G and H, regardless of their structure, precisely quantifying the dependence on t. In contrast, Theorem 1.3 shows that $E(G)$ samples, rather than $\log|G|$ samples, suffice to generate the entire domain group G and successfully apply the learn-and-test approach. Combining both aspects should yield stronger results in the online manipulation model, as in the case of finite fields (Theorem 1.5). More results of this flavor would be interesting, focusing on the following question: given a group G and $t \in \mathbb{N}$, how many random samples from G are required to generate w.h.p. a subgroup $G' \subseteq G$ of size $|G'|$ that exceeds t?

A different avenue is utilizing the structure of H and its *compatibility* to G, both for upper and lower bounds. As mentioned in Remark 1.7, H could reduce the complexity dramatically when it is incompatible with G. Is it possible to define a measure of compatibility that dictates the complexity of testing homomorphism from G to H? For lower bounds, could we find for each G a group H so that Theorem 1.3 is tight? One candidate would be $H = Aut(G)$, the group of automorphisms from G to G.

Organization. The rest of the paper is organized as follows. We begin in Section 2 with formal definitions of the problem and the online manipulation model. In Section 3, we introduce our core technical contribution, the Random Signs Test (Algorithm 1), and analyze its soundness in the standard (non-adversarial) model. In Section 4, we present the general framework for achieving online manipulation resilience, defining *unpredictable testers* and proving our general amplification lemma (Lemma 4.4). We then apply this framework in Section 5 to prove our main optimal bound for general groups (Theorem 1.1). Finally, our group-specific results, including the bounds for sample-based testers (Theorem 1.3) and for functions over prime fields (Theorems 1.5 and 1.6) are deferred to the full version.

2 Preliminaries

For two groups $(G,+)$ and $(H,\oplus)$, the function $h : G \to H$ is a group *homomorphism* if $h(x_1+x_2) = h(x_1) \oplus h(x_2)$ for all $x_1, x_2 \in G$. In non-Abelian groups, the order matters. The terms in the sums are ordered according to the index: in increasing index order for $\sum_{i\in[k]} x_i$ or $\bigoplus_{i=1}^{k} y_i$, and in decreasing index order for $\sum_{i=k}^{1} x_i$. The set of all group homomorphisms from G to H is denoted $\mathsf{HOM}(G,H)$. The (relative) Hamming *distance* between two functions $f, g : G \to H$ is $dist(f,g) := \Pr_{x\in G}[f(x) \neq g(x)]$. The *distance from* $f : G \to H$ *to a property* (i.e., a set) $\mathcal{P}$ of functions of the same form is $dist(f,\mathcal{P}) := \inf_{g\in\mathcal{P}} dist(f,g)$. For every function $f : G \to H$, let $\varepsilon_f := dist(f, \mathsf{HOM}(G,H))$. A function f is *ε-close to a homomorphism* if $\varepsilon_f \leq \varepsilon$ and is *ε-far from a homomorphism* otherwise.

▶ **Definition 2.1** (Online ε-tester [31, 10]). *Fix $\varepsilon \in (0,1)$. An online ε-tester $\mathcal{T}$ for a property $\mathcal{P}$ that works in the presence of a specified adversary (e.g., t-online erasure budget-managing) is given access to an input function f via oracle access to a related function f' that is initially equal to f and, after each query answered, may be modified by the adversary. For all adversarial strategies of the specified type,*

1. *if $f \in \mathcal{P}$, then $\mathcal{T}$ accepts with probability at least 2/3, and*
2. *if f is ε-far from $\mathcal{P}$, then $\mathcal{T}$ rejects with probability at least 2/3,*

where the probability is over the internal randomness of $\mathcal{T}$. If $\mathcal{T}$ works in the presence of an erasure (resp., corruption) adversary, we refer to it as an online-erasure-resilient (resp., online-corruption-resilient) tester. If $\mathcal{T}$ always accepts every function $f \in \mathcal{P}$, then $\mathcal{T}$ has 1-sided error. *If all queries of $\mathcal{T}$ are chosen uniformly and independently from the domain of f, then $\mathcal{T}$ is* sample-based.

3 Random Signs Test

In this section, we present our Random Signs Test (Algorithm 1) for testing group homomorphism. This test always accepts functions that are homomorphisms because it only rejects if a violation of the homomorphism property is found. We analyze the soundness of the test in Theorem 3.1.

Algorithm 1 Random Signs Test$_k$.

Parameters: $k \in \mathbb{N}$

Input: Query access to a function $f : G \to H$ where $(G, +)$ and $(H, \oplus)$ are finite groups.

1: Draw k independent and uniformly random elements $x_1, x_2, \ldots, x_k$ from G.
2: Draw k independent and uniformly random signs $\sigma_1, \ldots, \sigma_k$ from $\{+, -\}$.
3: Query $f(x_1), f(x_2), \ldots, f(x_k)$ and $f(a)$, where $a \leftarrow \sum_{i\in[k]} \sigma_i x_i$.
▷ $+b = b$ and $-b$ is the inverse of b for both $b \in G$ and $b \in H$.
4: **Reject** if $\bigoplus_{i\in[k]} \sigma_i f(x_i) \neq f(a)$; otherwise, **accept**.

▶ **Theorem 3.1.** *For all $k \in \mathbb{N}$, $\varepsilon \in (0,1)$, finite groups $(G, +)$ and $(H, \oplus)$, and functions $f : G \to H$ such that f is ε-far from being a homomorphism,*

$$\Pr\left[\text{Random Signs Test}_{2k} \text{ rejects}\right] \geq \min\left\{\frac{(2k-3)\cdot\varepsilon}{3}, \frac{1}{16}\right\}. \tag{3}$$

Recall that ε_f denotes the (relative) distance from f to $\mathsf{HOM}(G, H)$. We analyze the soundness of Random Signs Test for small ε_f in Section 3.1 and for large ε_f in Section 3.2. We put these two results together and complete the proof of Theorem 3.1 in Section 3.3.

3.1 Soundness analysis for small ε_f

In this section, we state and prove Lemma 3.2, establishing the soundness of Algorithm 1 for $\varepsilon_f < \frac{1}{8}$.

▶ **Lemma 3.2.** *For all $k \in \mathbb{N}$, finite groups $(G, +)$ and $(H, \oplus)$, and functions $f : G \to H$ with $\varepsilon_f < \frac{1}{8}$,*

$$\Pr\left[\text{Random Signs Test}_{k} \text{ rejects}\right] \geq \min\left\{\frac{(k-3)\cdot\varepsilon_f}{3}, \frac{1}{6}\right\}. \tag{4}$$

To prove Lemma 3.2, we introduce Fixed Test$_k$ (Algorithm 2) which uses a fixed sequence of k signs $\overline{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_k)$, in contrast to the uniform and independent signs from $+, -$ used in Algorithm 1.

Algorithm 2 Fixed Test$_k(\overline{\sigma})$.

Parameters: $k \in \mathbb{N}$, sign sequence $\overline{\sigma} \in \{+,-\}^k$
Input: Query access to a function $f : G \to H$ where $(G,+)$ and $(H,\oplus)$ are finite groups.
1: Draw k independent and uniformly random elements $x_1, x_2, \ldots, x_k$ from G.
2: Query $f(x_1), f(x_2), \ldots, f(x_k)$ and $f(a)$, where $a \leftarrow \sum_{i\in[k]} \sigma_i x_i$.
3: **Reject** if $\bigoplus_{i\in[k]} \sigma_i f(x_i) \neq f(a)$; otherwise **accept**.

Note that Algorithm 2 with $\overline{\sigma}$ set to $+^k$ recovers the tester of [31].

▶ **Lemma 3.3.** *For all $k \in \mathbb{N}$, all sign sequences $\overline{\sigma} \in \{+,-\}^k$, finite groups $(G,+)$ and $(H,\oplus)$, and functions $f : G \to H$ with $\varepsilon_f < \frac{1}{8}$,*

$$\Pr\left[\text{Fixed Test}_k(\overline{\sigma}) \text{ rejects}\right] \geq \min\left\{\frac{(k-3)\cdot\varepsilon_f}{3}, \frac{1}{6}\right\}. \tag{5}$$

We prove Lemma 3.3 after showing that it implies Lemma 3.2.

Proof of Lemma 3.2 assuming Lemma 3.3. Random Signs Test$_k$ (Algorithm 1) chooses the signs $\overline{\sigma}$ independently and uniformly at random from $\{+,-\}^k$. Thus, the rejection probability of Random Signs Test$_k$ is the expected rejection probability of Fixed Test$_k(\overline{\sigma})$ for uniformly random $\overline{\sigma}$. ◀

Proof of Lemma 3.3. Fix a sign sequence $\overline{\sigma}$ in $\{+,-\}^k$. Let $g : G \to H$ be a closest homomorphism to f, i.e., $\Pr_{x\in G}[f(x) \neq g(x)] = \varepsilon_f$. Then Fixed Test$_k(\overline{\sigma})$ always accepts g, since for all $x_1, x_2, \ldots, x_k \in G$,

$$\bigoplus_{i\in[k]} \sigma_i g(x_i) = g\Big(\sum_{i\in[k]} \sigma_i x_i\Big). \tag{6}$$

The core of the analysis is to compare how Algorithm 2 behaves on f and g for each sample. One way Algorithm 2 can reject f is by selecting $x_1, \ldots, x_k$ such that the RHS of (6) remains unchanged when replacing g with f, but the LHS does not.

$$\Pr\left[\text{Fixed Test}_k(\overline{\sigma}) \text{ rejects}\right] \geq \Pr_{x_1,\ldots,x_k\in G}\Big[\bigoplus_{i\in[k]} \sigma_i f(x_i) \neq \bigoplus_{i\in[k]} \sigma_i g(x_i) \bigwedge f\Big(\sum_{i\in[k]} \sigma_i x_i\Big) = g\Big(\sum_{i\in[k]} \sigma_i x_i\Big)\Big]. \tag{7}$$

We consider the probabilities of the complements of the two events – on the LHS and the RHS of (7) – obtaining the complement of the event on the RHS by applying the De Morgan's law:

$$\begin{aligned}
&\Pr[\text{Fixed Test}_k(\overline{\sigma}) \text{ accepts}] \\
&\leq \Pr_{x_1,\ldots,x_k\in G}\Big[\bigoplus_{i\in[k]} \sigma_i f(x_i) = \bigoplus_{i\in[k]} \sigma_i g(x_i) \bigvee f\Big(\sum_{i\in[k]} \sigma_i x_i\Big) \neq g\Big(\sum_{i\in[k]} \sigma_i x_i\Big)\Big] \\
&\leq \Pr\Big[\bigoplus_{i\in[k]} \sigma_i f(x_i) = \bigoplus_{i\in[k]} \sigma_i g(x_i)\Big] + \Pr\Big[f\Big(\sum_{i\in[k]} \sigma_i x_i\Big) \neq g\Big(\sum_{i\in[k]} \sigma_i x_i\Big)\Big],
\end{aligned} \tag{8}$$

where (8) holds by a union bound. The second term of (8) is exactly ε_f, as $\sum_{i\in[k]}\sigma_i x_i$ is a uniformly random element from G. Let p_k denote the first term of (8) where the subscript k indicates the number of elements the probability is taken over.

We next use induction on k to prove the following. Let $k_0 = \lfloor 1/\varepsilon_f \rfloor$, then

$$p_k \le \begin{cases} 1 - k\varepsilon_f/e & \text{for } k \in [k_0]; \\ \max\{p_{k_0}, 1/2 + \varepsilon_f\} & \text{for } k > k_0. \end{cases}$$

▷ Claim 3.4 (Base Case). For all $k \le 1/\varepsilon_f$, we have $p_k \le 1 - k\varepsilon_f/e$.

▷ Claim 3.5 (Inductive Step). For all $k > 1$, we have $p_k \le \max\{p_{k-1}, 1/2 + \varepsilon_f\}$.

The proofs of Claim 3.4 and Claim 3.5 are deferred to the full version.

Note that $k_0 = \lfloor 1/\varepsilon_f \rfloor > 1/\varepsilon_f - 1$. Using Claim 3.4 we have $p_{k_0} \le 1 - k_0\varepsilon_f/e < 1 - (1-\varepsilon_f)/e$. If $\varepsilon_f \le 1/8$, then $p_{k_0} \le 0.7$. Using Claim 3.5 we have for all $k \ge k_0$, $p_k \le \max\{p_{k_0}, 1/2 + \varepsilon_f\} \le 0.7$ for $\varepsilon_f \le 1/8$. We complete the proof by noting that $\Pr[\textsf{Fixed Test}_k(\overline{\sigma}) \text{ accepts}] \le p_k + \varepsilon_f$ from (8). Thus, $\Pr[\textsf{Fixed Test}_k(\overline{\sigma}) \text{ rejects}] \ge (1 - p_k) - \varepsilon_f \ge \min\{k\varepsilon_f/e - \varepsilon_f, 0.3 - \varepsilon_f\} \ge \min\{(k-3)\varepsilon_f/3, 1/6\}$. ◀

3.2 Soundness analysis for large ε_f

In this section, we show that Random Signs Test used with *even* k provides soundness proportional to ε_f. Specifically, we prove the following lemma.

▶ **Lemma 3.6.** *For all $k \in \mathbb{N}$, finite groups $(G, +)$ and $(H, \oplus)$, and functions $f : G \to H$,*

$$\Pr[\textsf{Random Signs Test}_{2k} \textit{ rejects}] \ge \min\left\{\frac{\varepsilon_f}{2}, \frac{1}{10}\right\}. \tag{9}$$

For ease of notation, let us define $\mu := \Pr[\textsf{Random Signs Test}_{2k} \text{ rejects } f]$. Lemma 3.6 immediately follows from the following lemma.

▶ **Lemma 3.7.** *If $\mu < \frac{1}{10}$, then f is 2μ-close to a homomorphism.*

Before proving Lemma 3.7, we explain why it implies Lemma 3.6. By Lemma 3.7, if $\mu < 1/10$ then f is 2μ-close to some homomorphism. Since f has distance ε_f to $\textsf{HOM}(G, H)$, we get $2\mu \ge \varepsilon_f \implies \mu \ge \frac{\varepsilon_f}{2}$. Thus, $\mu \ge \min\left\{\frac{\varepsilon_f}{2}, \frac{1}{10}\right\}$. Next we prove Lemma 3.7.

Proof of Lemma 3.7. In the test, the distribution of the element a, as specified in Algorithm 2, Line 2, is uniform over all elements of G, since for each realization of the signs $\sigma_1, \ldots, \sigma_{2k}$ and $x_1, \ldots, x_{2k-1}$, there is a single choice of x_{2k} leading to each a. Define $\text{FIX}(a)$ to be the set of realizations for σ's and x's leading to a,

$$\text{FIX}(a) := \left\{(\sigma_1, \ldots, \sigma_{2k}, x_1, \ldots, x_{2k}) : \textstyle\sum_{i\in[2k]} \sigma_i x_i = a\right\}.$$

We use $(\overline{\sigma}, \overline{x})$ as a shorthand for $(\sigma_1, \ldots, \sigma_{2k}, x_1, \ldots, x_{2k})$. A random tuple from $\text{FIX}(a)$ is k-wise independent in the following sense.

▷ Claim 3.8. For all $a \in G$, all sets of indices $S \subseteq [2k]$ of size $|S| = k$, let $(\overline{\sigma}, \overline{x})$ be a uniformly random tuple from $\text{FIX}(a)$. Then

1. The vector $(\sigma_i, x_i)_{i\in S}$ is a uniformly distributed vector of k signs and k elements from G.
2. The partial sum $\sum_{i\in S} \sigma_i x_i$ is a uniformly distributed element in G.

Proof.

Part 1. Consider a tuple $(\sigma_i, x_i)_{i\in[2k]}$, not necessarily in $\mathrm{FIX}(a)$, and fix a partial assignment that gives values only to $(\sigma_i, x_i)_{i\in S}$. Now choose an index $j \in [2k] \setminus S$ and let (σ_j, x_j) be chosen uniformly at random. Fill up the remaining indices, (σ_i, x_i) for $i \in [2k] \setminus (S \cup \{j\})$ with arbitrary values. Since (σ_j, x_j) is chosen uniformly at random, the sum $\sum_{i=1}^{j} \sigma_i x_i$ is a uniformly random element in G. Also, for any group G, the function $x \to x + g$ is a bijection from G to G for all $g \in G$ (the same holds true for $g + x, x - g, -g + x$). Thus, the rest of the additions and subtractions (for $i = j + 1, \ldots, 2k$) are all bijections and therefore the entire sum $\sum_{i\in[2k]} \sigma_i x_i$ is also a uniformly random element of G. This means that exactly $1/|G|$ of the tuples with a particular partial assignment to $(\sigma_i, x_i)_{i\in S}$ are in fact in $\mathrm{FIX}(a)$. The same argument applies for any partial assignment of $(\sigma_i, x_i)_{i\in S}$, so each such partial assignment produces the exact same number of tuples in $\mathrm{FIX}(a)$.

Part 2. Consider the set F_z which consists of all assignments to $(\sigma_i, x_i)_{i\in S}$ for which $\sum_{i\in S} \sigma_i x_i = z$. Using similar arguments as in the proof of part 1, each set F_z has the same size, and the same number of ways to complete the assignment to a tuple in $\mathrm{FIX}(a)$. Thus, the number of tuples in $\mathrm{FIX}(a)$ for which $\sum_{i\in S} \sigma_i x_i = z$ is the same for each z. ◁

Consider tuples whose signed sums over either their left or right halves equal a fixed value. Define

$$\mathrm{LEFT}(z) := \left\{(\overline{\sigma}, \overline{x}) : \textstyle\sum_{i\in[k]} \sigma_i x_i = z\right\}, \quad \mathrm{RIGHT}(z) := \left\{(\overline{\sigma}, \overline{x}) : \textstyle\sum_{i\in[k]} \sigma_{k+i} x_{k+i} = z\right\}.$$

▶ **Corollary 3.9.** *Consider the sets $S_L = [k]$ and $S_R = [2k] \setminus [k]$. The following process produces a uniformly random tuple in FIX(a):*

1. *Draw a uniformly random element $z \in G$.*
2. *Draw a uniformly random tuple $(\overline{\sigma}^L, \overline{x}^L)$ from LEFT$(a - z)$ and take its "left" part, $(\overline{\sigma}^L, \overline{x}^L)_{i\in S_L}$*
3. *Draw a uniformly random tuple $(\overline{\sigma}^R, \overline{x}^R)$ from RIGHT(z) and take its "right" part, $(\overline{\sigma}^R, \overline{x}^R)_{i\in S_R}$*
4. *Return the tuple $(\overline{\sigma}, \overline{x})$ where $(\overline{\sigma}, \overline{x})_{i\in S_L} = (\overline{\sigma}^L, \overline{x}^L)_{i\in S_L}$ and $(\overline{\sigma}, \overline{x})_{i\in S_R} = (\overline{\sigma}^R, \overline{x}^R)_{i\in S_R}$*

Proof. By the second part of Claim 3.8, the number of tuples $(\sigma_1, \ldots, \sigma_{2k}, x_1, \ldots, x_{2k}) \in \mathrm{FIX}(a)$ such that their right part is in $\mathrm{RIGHT}(z)$ (that is, $\sum_{i\in[k]} \sigma_{k+i} x_{k+i} = z$) is the same for each z. The same holds for the $\mathrm{LEFT}(a - z)$. Since each tuple in $\mathrm{FIX}(a)$ has a unique such pair $(a - z, z)$, the process gives an equal probability for each tuple in $\mathrm{FIX}(a)$, concluding the proof. ◀

▶ **Definition 3.10** (Vote of $(\overline{\sigma}, \overline{x})$ for the value of a)**.** *For all $a \in G$ and $(\overline{\sigma}, \overline{x}) \in$ FIX(a), define $\bigoplus_{i\in[2k]} \sigma_i f(x_i)$ as the* vote of $(\overline{\sigma}, \overline{x})$ for the value of a.

Corrector. Define g to be the local corrector for f, that is, the value that has the most votes as defined in Definition 3.10:

$$g(a) := \mathrm{argmax}_{h\in H} \left\{\Pr_{(\overline{\sigma},\overline{x})\in \mathrm{FIX}(a)} \left[\textstyle\bigoplus_{i\in[2k]} \sigma_i f(x_i) = h\right]\right\} \text{ for all } a \in G.$$

Define the following probabilities, all relating to how close f is to being homomorphic:

$$\eta_a := \Pr_{(\overline{\sigma},\overline{x})\in \mathrm{FIX}(a)}\left[g(a) \neq \bigoplus_{i\in[2k]} \sigma_i f(x_i)\right] \quad \eta := max_a \eta_a \quad \delta := \Pr_{x\in G}\left[f(x) \neq g(x)\right]$$

We follow the standard self-correction paradigm introduced by [15, 11], by showing that if the test fails with a small probability (e.g., $\mu < 1/10$), then: (1) η is bounded; (2) $\delta \leq 2\mu$; and (3) g is a homomorphism. The last part (3) is evidently the harder one, although similar simpler arguments are used to prove (1). Altogether, they show that if the test fails with a small probability, then f is close to being a homomorphism (and in particular, the homomorphism g). By contrapositive, this proves a soundness guarantee for the tester.

We start by formally stating and proving the first assertion.

▶ **Lemma 3.11.** $\eta \leq 2\mu$.

Proof. It suffices to show that $\eta_a \leq 2\mu$ for each $a \in G$. Fix some a and consider two tuples $(\overline{\sigma},\overline{x}), (\overline{\tau},\overline{y})$ chosen independently and uniformly at random from $\mathrm{FIX}(a)$. We show that the probability of the event that the votes of $(\overline{\sigma},\overline{x})$ and $(\overline{\tau},\overline{y})$ for the value of a are equal is close to 1, which implies the most common vote among all tuples in $\mathrm{FIX}(a)$ occurs with probability close to 1. To check whether they give the same vote we manipulate both summations as follows: apply from the left side $-\tau_k f(y_k)\cdots-\tau_1 f(y_1)$ to cancel out the first k summands from the tuple $(\overline{\tau},\overline{y})$, and similarly from the right side for the last k summands of the other tuple. We get the following equivalence:

$$\underbrace{\bigoplus_{i\in[2k]} \sigma_i f(x_i)}_{S_{\sigma,x}} = \underbrace{\bigoplus_{i\in[2k]} \tau_i f(y_i)}_{S_{\tau,y}} \iff$$

$$\underbrace{\bigoplus_{i=0}^{k-1} -\tau_{k-i} f(y_{k-i}) \oplus \bigoplus_{i\in[k]} \sigma_i f(x_i)}_{S_1} = \underbrace{\bigoplus_{i=k+1}^{2k} \tau_i f(y_i) \oplus \bigoplus_{i=0}^{k-1} -\sigma_{2k-i} f(x_{2k-i})}_{S_2}.$$

Importantly, each of the new sums S_1, S_2 contains exactly k summands respectively from $(\overline{\sigma},\overline{x})$ and $(\overline{\tau},\overline{y})$, but with different signs. By Claim 3.8 and independence between $(\overline{\sigma},\overline{x})$ and $(\overline{\tau},\overline{y})$, the tuple of $2k$ signs and $2k$ elements used in S_1 is uniformly random (and similarly for S_2). Since the original tuples are taken from $\mathrm{FIX}(a)$, $\sum_{i\in[2k]} \sigma_i x_i = a = \sum_{i\in[2k]} \tau_i y_i$. By the same reasoning as before but with x_i, y_i instead of $f(x_i), f(y_i)$

$$\textstyle\sum_{i=0}^{k-1} -\tau_{k-i} y_{k-i} + \sum_{i\in[k]} \sigma_i x_i = \sum_{i=k+1}^{2k} \tau_i y_i + \sum_{i=0}^{k-1} -\sigma_{2k-i} x_{2k-i}.$$

Thus, the two tuples defining the sums or votes S_1, S_2 are both in $\mathrm{FIX}(a')$, for the same uniformly random element $a' \in G$. The event $\{S_1 \neq S_2\}$ implies that at least one of the votes did not agree with $f(a')$.

$$\Pr\left[S_1 \neq S_2\right] \leq \Pr\left[S_1 \neq f(a') \vee S_2 \neq f(a')\right] \leq \Pr\left[S_1 \neq f(a')\right] + \Pr\left[S_2 \neq f(a')\right] = 2\mu,$$

where the second inequality is by a union bound, and the last equality holds since both the probabilities are equal to the probability that the check in Line 4 of Algorithm 1 accepts, thus simulating our test. Indeed, $S_i \in \mathrm{FIX}(a')$ for $i = 1, 2$, where a' is a uniformly random element of G. We denote by p_h the probability that a correction using a random tuple $(\overline{\sigma},\overline{x}) \in \mathrm{FIX}(a)$ produces $h \in H$, then:

$$2\mu \geq \Pr\left[S_1 \neq S_2\right] = \Pr\left[S_{\sigma,x} \neq S_{\tau,y}\right] = \sum_{h\in H} p_h(1-p_h) = \sum_{h\in H} (p_h - p_h^2) = 1 - \sum_{h\in H} p_h^2.$$

By definition, $g(a) = \arg\max_{h \in H} p_h$ is the most common vote or correction outcome (ties broken arbitrarily). We have $1 - \eta_a = p_{g(a)} = p_{g(a)} \cdot \sum_{h \in H} p_h = \sum_{h \in H} p_h p_{g(a)} \geq \sum_{h \in H} p_h^2 \geq 1 - 2\mu$, which concludes the proof. ◀

The next simple lemma shows the second easy assertion as a corollary of the first.

▶ **Lemma 3.12.** *If $\mu < 1/10$, then $\delta \leq 2\mu$.*

Proof. For each element $a \in G$ we have high agreement for corrections, and in particular $\eta_a \leq 1/2$. Denote the set where f and g disagree by $\Delta := \{x \in G : f(x) \neq g(x)\}$, noting that $\delta = |\Delta| / |G|$. Then

$$\mu = \Pr_{\substack{a \in G \\ (\sigma,x) \in \mathrm{FIX}(a)}} \left[f(a) \neq \bigoplus_{i \in [2k]} \sigma_i f(x_i) \right] = \frac{1}{|G|} \sum_{a \in G} \Pr_{(\sigma,x) \in \mathrm{FIX}(a)} \left[f(a) \neq \bigoplus_{i \in [2k]} \sigma_i f(x_i) \right].$$

The latter expression is only reduced if we take a subset of (positive) summands, only those from $\Delta \subseteq G$. It further reduces if we replace each event of the sum differing from $f(a)$ with the subevent that it specifically equals $g(a)$ (this is indeed a subevent for any $a \in \Delta$). We get

$$\mu \geq \frac{1}{|G|} \sum_{a \in \Delta} \Pr_{(\sigma,x) \in \mathrm{FIX}(a)} \left[g(a) = \bigoplus_{i \in [2k]} \sigma_i f(x_i) \right] = \frac{1}{|G|} \sum_{a \in \Delta} (1 - \eta_a) \geq \frac{1}{|G|} \cdot |\Delta| \cdot \frac{1}{2} = \frac{\delta}{2},$$

where the last inequality uses $1 - \eta_a \geq 1/2$ for any $a \in \Delta$. This concludes the proof. ◀

The last assertion uses the probabilistic method and the accuracy of corrections shown above, to prove that g is a homomorphism.

▶ **Lemma 3.13.** *If $\mu < 1/10$, then g is a homomorphism.*

Proof. Given $a, b \in G$, we need to show that $g(a) \oplus g(b) = g(a + b)$. Consider the following process:

1. Draw a uniformly random $r \in G$.
2. Draw a uniformly random tuple $(\overline{\sigma}, \overline{x}) \in \mathrm{LEFT}(a - r) \bigcap \mathrm{RIGHT}(r)$.
3. Let CL be the set "cancelling left" whose left side cancels exactly the right side of the previous tuple, defined as, $CL := \{(\overline{\tau}, \overline{y}) : \forall i \in [k].(\tau_i, y_i) = (-\sigma_{2k+1-i}, x_{2k+1-i})\}$, and choose uniformly at random a tuple $(\overline{\tau}, \overline{y})$ from $CL \bigcap \mathrm{RIGHT}(r + b)$.

Applying Corollary 3.9 three times, note that $(\overline{\sigma}, \overline{x})$ is a uniformly random tuple in $\mathrm{FIX}(a)$, the tuple $(\overline{\tau}, \overline{y})$ is a uniformly random tuple in $\mathrm{FIX}(b)$. Furthermore, we define the tuple $(\overline{\phi}, \overline{z})$ that has the left side taken from $(\overline{\sigma}, \overline{x})$ and the right side taken from $(\overline{\tau}, \overline{y})$, that is $(\phi_i, z_i) = (\sigma_i, x_i)$ for $i \in [k]$ and $(\phi_i, z_i) = (\tau_i, y_i)$ for $i \in [2k] \setminus [k]$. Applying Corollary 3.9 again, we get that $(\overline{\phi}, \overline{z})$ is a uniformly random tuple from $\mathrm{FIX}(a + b)$. Hence, each of the following three equalities holds with probability at least $1 - \eta$:

$$g(a) = \bigoplus_{i \in [2k]} \sigma_i f(x_i) \qquad g(b) = \bigoplus_{i \in [2k]} \tau_i f(y_i) \qquad g(a+b) = \bigoplus_{i \in [2k]} \phi_i f(z_i)$$

Next, recall we specifically chose the left side of $(\overline{\tau}, \overline{y})$ to cancel out the right side of $(\overline{\sigma}, \overline{x})$. This cancellation also happens when applying the function f, since pairs of summands cancel each other from inside out. The following equality thus holds with probability 1:

$$\bigoplus_{i=k+1}^{2k} \sigma_i f(x_i) \oplus \bigoplus_{i \in [k]} \tau_i f(y_i) = \sigma_{k+1} f(x_{k+1}) \oplus \cdots \oplus \sigma_{2k} f(x_{2k}) \oplus \tau_1 f(y_1) \oplus \cdots \oplus \tau_k f(y_k) = e_H.$$

Over the random choice of tuples, the event that all 4 equalities above hold has probability at least $1-3\eta \geq 1-6\mu > 0$ (by applying Lemma 3.11, a union bound, and the premise $\mu < 1/6$). This event implies that $g(a) \oplus g(b) = g(a+b)$, which also occurs with positive probability. However, the last equality does not depend on the probability space, and therefore it must hold with probability 1. ◀

This completes the proof of Lemma 3.7. ◀

3.3 Completing the soundness analysis

Proof of Theorem 3.1. Since f is ε-far from being a homomorphism, $\varepsilon_f \geq \varepsilon$. When $\varepsilon_f \geq 1/8$, then by Lemma 3.6, we get $\Pr[\textsf{Random Signs Test}_{2k} \text{ rejects}] \geq \min\left\{\frac{\varepsilon_f}{2}, \frac{1}{10}\right\} \geq \frac{1}{16}$. On the other hand, if $\varepsilon_f < 1/8$, then by Lemma 3.2, $\Pr[\textsf{Random Signs Test}_{2k} \text{ rejects}] \geq \min\left\{\frac{(2k-3)\cdot\varepsilon_f}{3}, \frac{1}{6}\right\} \geq \min\left\{\frac{(2k-3)\cdot\varepsilon}{3}, \frac{1}{6}\right\}$. Combining the two cases, we get

$$\Pr[\textsf{Random Signs Test}_{2k} \text{ rejects}] \geq \min\left\{\frac{(2k-3)\cdot\varepsilon}{3}, \frac{1}{6}, \frac{1}{16}\right\} \geq \min\left\{\frac{(2k-3)\cdot\varepsilon}{3}, \frac{1}{16}\right\}. \quad ◀$$

4 General framework for online manipulation resilience

The previous section provides a sound *core test* for homomorphism. To make this core test resilient to online manipulations, we now introduce the general framework used to build our final testers. This framework, based on the concept of *unpredictability* and an amplification lemma, provides a general recipe for converting a core test with certain unpredictability properties, i.e., an *unpredictable core test*, to one robust to online manipulations. In subsequent sections, we show how to modify our core test into an unpredictable core test and then apply this framework to prove our main results.

4.1 Unpredictable test

We define flatness of a distribution, which can also be seen as a bound on its "min-entropy".[8]

▶ **Definition 4.1** (α-flat). *A distribution ν on domain D is α-flat if* $\max_{z\in D} \nu(z) \leq \alpha$.

The following definition, loosely speaking, requires that other than with probability β of "losing its unpredictability", a q-query test maintains "aggregated flatness" of α. Later in Lemma 4.4, we show that such a tester is resilient to online manipulations.

▶ **Definition 4.2** (Unpredictable tester). *A tester making q sequential queries $x_1, \ldots, x_q$ to a function $f : D \to R$ is* ***(q, α, β)-unpredictable*** *(in the standard model, without an adversary) if there exist nonnegative flatness parameters $(\alpha_1, \ldots, \alpha_q)$ and failure probabilities $(\beta_1, \ldots, \beta_q)$ satisfying the following two conditions.*

1. ***Sequential unpredictability:*** *For every input function $f : D \to R$, the following holds. For each $j \in [q]$, let ν_j be the probability distribution of the j-th query x_j conditioned on all the previous queries $(x_1, \ldots, x_{j-1})$. Then, for each $j \in [q]$, with probability at least $1-\beta_j$ (taken over the randomness of the previous $j-1$ queries), ν_j is α_j-flat:*

$$\Pr_{x_1,\ldots,x_{j-1}}[\nu_j \text{ is } \alpha_j\text{-flat}] \geq 1-\beta_j.$$

2. ***Aggregate parameters:*** *The sums of flatness parameters and of the failure probabilities are bounded, satisfying $\sum_{j\in[q]} \alpha_j \leq \alpha$ and $\sum_{j\in[q]} \beta_j \leq \beta$.*

[8] A distribution is α-flat if and only if its min-entropy is at least $\log(1/\alpha)$.

▶ **Lemma 4.3** (Encountering a manipulation). *Let $\mathcal{A}$ be an algorithm that runs r independent iterations of a (q, α, β)-unpredictable test. Fix an iteration $i \in [r]$, and let E_i be an event completely determined by the first $i-1$ iterations. Then, for all input functions f and t-online budget-managing adversarial strategies, the probability that a manipulation is encountered in iteration i, conditioned on the event E_i, is at most $\alpha qrt+\beta$. (Note that qrt is the manipulation budget for the entire algorithm.)*

Proof. Fix an adversarial strategy. Since $\mathcal{A}$ runs a (q, α, β)-unpredictable test, there exist flatness parameters $(\alpha_1, \ldots, \alpha_q)$ and error terms $(\beta_1, \ldots, \beta_q)$ satisfying the conditions in Definition 4.2. Let M_i be the event that a manipulation was encountered in iteration i. For all $j \in [q]$, let M_{ij} be the event that the j-th query of iteration i is to a manipulated point.

We now bound the probability $\Pr[M_{ij}|E_i]$ for a single i. To do this, consider the sample space of all execution histories of $\mathcal{A}$ up to iteration $i-1$. The event E_i is simply a collection of such histories. Fix a specific execution history h, which fixes all queries made by $\mathcal{A}$ and all manipulations made by the adversary in the previous $i-1$ iterations.

Let G_j be the good event that the distribution ν_j is α_j-flat. Then $\Pr[\overline{G_j}|h] \le \beta_j$. Suppose the event G_j occurred, and let $\mathcal{M}_j$ be the set of manipulated entries when query j is made. Then $|\mathcal{M}_j| \le qrt$. Thus, the probability of the event M_{ij} conditioned on G_j and h,

$$\Pr[M_{ij} \mid G_j, h] = \sum_{z \in \mathcal{M}_j} \Pr_{x_j \sim \nu_j}[x_j = z] \le \sum_{z \in \mathcal{M}_j} \alpha_j \le \alpha_j qrt.$$

The probability of the event M_{ij} conditioned on the history h is bounded by

$$\begin{aligned}\Pr[M_{ij} \mid h] &= \Pr[M_{ij} \mid G_j, h]\Pr[G_j \mid h] + \Pr[M_{ij} \mid \overline{G_j}, h]\Pr[\overline{G_j} \mid h] \\ &\le \Pr[M_{ij} \mid G_j, h] + \Pr[\overline{G_j} \mid h] \le \alpha_j qrt + \beta_j.\end{aligned}$$

By a union bound over the events M_{ij} for all $j \in [q]$, the probability of the event M_i conditioned on h is

$$\Pr[M_i \mid h] \le \sum_{j \in [q]} \Pr[M_{ij}|h] = \sum_{j \in [q]} \left(\alpha_j qrt + \beta_j\right) \le \alpha qrt + \beta.$$

Now, consider all execution histories that satisfy the event E_i.

$$\begin{aligned}\Pr[M_i \mid E_i]\Pr[E_i] = \Pr[M_i \cap E_i] &= \sum_{h \in E_i} \Pr[M_i \cap h] = \sum_{h \in E_i} \Pr[M_i|h]\Pr[h] \\ &\le (\alpha qrt + \beta) \cdot \sum_{h \in E_i} \Pr[h]\end{aligned}$$

$$\implies \Pr[M_i \mid E_i]\Pr[E_i] \le (\alpha qrt + \beta) \cdot \Pr[E_i] \implies \Pr[M_i \mid E_i] \le \alpha qrt + \beta. \qquad \blacktriangleleft$$

4.2 General amplification lemma

In this section, we prove a general amplification lemma. This result provides a recipe for converting an unpredictable core test into an online manipulation-resilient tester. Specifically, we show that if a (q, α, β)-unpredictable test satisfies a condition linking its soundness $p_{\mathsf{rej}}(\varepsilon)$ (probability of rejecting $f \notin \mathcal{P}$ as a function of ε) to its aggregate unpredictability parameters (α, β) and the adversary's manipulation rate t, then running the test $O(1/p_{\mathsf{rej}}(\varepsilon))$ times yields a new tester with a constant soundness guarantee against the adversary.

▶ **Lemma 4.4** (General Amplification for Online Resilience). *Let $\mathcal{P}$ be a property (a set of functions $f : D \to R$). Let $\mathcal{T}$ be a (q, α, β)-unpredictable, 1-sided error ε-test for property $\mathcal{P}$. Suppose $\mathcal{T}$ rejects all functions that are ε-far from $\mathcal{P}$ with probability at least $p_{\mathsf{rej}}(\varepsilon)$. Let $\mathcal{A}$ be the algorithm that runs $\lceil 3/p_{\mathsf{rej}}(\varepsilon)\rceil$ independent iterations of test $\mathcal{T}$ and accepts iff each iteration accepts or (in the case when manipulations are erasures) encounters an erasure. Algorithm $\mathcal{A}$ is an ε-tester for property $\mathcal{P}$ that works in the presence of every t-online manipulation budget-managing adversary if*

$$\alpha q t \cdot \left\lceil \frac{3}{p_{\mathsf{rej}}(\varepsilon)} \right\rceil + \beta \leq \frac{p_{\mathsf{rej}}(\varepsilon)}{12}. \tag{10}$$

Moreover, if the manipulations are erasures, then $\mathcal{A}$ has 1-sided error.

Proof. Let $r = \lceil 3/p_{\mathsf{rej}}(\varepsilon)\rceil$ be the number of iterations of the test $\mathcal{T}$ run by $\mathcal{A}$.

Completeness. Consider $f \in \mathcal{P}$. The tester $\mathcal{A}$ fails on f only if one of the iterations rejects. This can only happen because of a manipulation, since $\mathcal{T}$ has 1-sided error (it always accepts $f \in \mathcal{P}$ when there are no manipulations). For all $i \in [r]$, let E_i be the trivial event that includes the entire sample space. By Lemma 4.3 instantiated with this E_i and by (10), the probability $\mathcal{A}$ encounters a manipulation in any single iteration is at most

$$\alpha q r t + \beta = \alpha q t \left\lceil \frac{3}{p_{\mathsf{rej}}(\varepsilon)} \right\rceil + \beta \leq \frac{p_{\mathsf{rej}}(\varepsilon)}{12}.$$

By a union bound over all $r = \lceil \frac{3}{p_{\mathsf{rej}}(\varepsilon)} \rceil$ iterations, $\mathcal{A}$ encounters a manipulation with probability at most

$$\frac{p_{\mathsf{rej}}(\varepsilon)}{12} \cdot \left\lceil \frac{3}{p_{\mathsf{rej}}(\varepsilon)} \right\rceil \leq \frac{1}{3}.$$

Therefore, $\mathcal{A}$ errs with probability at most $\frac{1}{3}$. Moreover, when manipulations are erasures, $\mathcal{A}$ always accepts f because it is designed to accept in any iteration where it sees an erasure.

Soundness. Assume f is ε-far from $\mathcal{P}$. Let F be the event that the amplified tester $\mathcal{A}$ fails to reject f. For all iterations $i \in [r]$, let F_i be the event that $\mathcal{A}$ fails on iteration i. Then $F = \bigcap_{i\in[r]} F_i$. Therefore,

$$\Pr[F] = \prod_{i\in[r]} \Pr[F_i \mid F_1 \cap \cdots \cap F_{i-1}]. \tag{11}$$

Next we bound each term in the product in (11). For all $i \in [r]$, let A_i be the event that $\mathcal{T}$ accepts in iteration i, and M_i be the event that a manipulated entry is queried in iteration i. Fix an iteration $i \in [r]$. Then $F_i \subseteq A_i \cup M_i = (A_i \cap \overline{M_i}) \cup M_i$. By a union bound,

$$\Pr[F_i \mid F_1 \cap \cdots \cap F_{i-1}] \leq \Pr[(A_i \cap \overline{M_i}) \mid F_1 \cap \cdots \cap F_{i-1}] + \Pr[M_i \mid F_1 \cap \cdots \cap F_{i-1}]. \tag{12}$$

The first term on the RHS of (12) corresponds to the event when $\mathcal{A}$ encounters no manipulations in iteration i and the unpredictable core test $\mathcal{T}$ accepts. In this case, the view of the algorithm $\mathcal{A}$ is the same as if the function f was not manipulated and conditioning on any event determined by the initial $i-1$ iterations such as $F_1 \cap \cdots \cap F_{i-1}$ doesn't affect the probability of the event $A_i \cap \overline{M_i}$. Since the core test $\mathcal{T}$ rejects f with probability at least $p_{\mathsf{rej}}(\varepsilon)$, this implies $\Pr[(A_i \cap \overline{M_i}) \mid F_1 \cap \cdots \cap F_{i-1}] \leq 1 - p_{\mathsf{rej}}(\varepsilon)$. To bound the second term on the RHS of (12), we use Lemma 4.3 with $E_i = \bigcap_{j\in[i-1]} F_j$ and then (10):

$$\Pr[M_i \mid F_1 \cap \cdots \cap F_{i-1}] \leq \alpha q r t + \beta = \alpha q t \left\lceil \frac{3}{p_{\mathsf{rej}}(\varepsilon)} \right\rceil + \beta \leq \frac{p_{\mathsf{rej}}(\varepsilon)}{12}.$$

By (12), $\Pr[F_i \mid F_1 \cap \cdots \cap F_{i-1}] \le (1 - p_{\mathsf{rej}}(\varepsilon)) + \frac{p_{\mathsf{rej}}(\varepsilon)}{12} = 1 - \frac{11 p_{\mathsf{rej}}(\varepsilon)}{12}$. The above bound holds for all iterations $i \in [r]$. Thus, by (11),

$$\Pr[F] \le \prod_{i \in [r]} \left(1 - \frac{11 p_{\mathsf{rej}}(\varepsilon)}{12}\right) = \left(1 - \frac{11 p_{\mathsf{rej}}(\varepsilon)}{12}\right)^r.$$

Substituting $r = \lceil 3/p_{\mathsf{rej}}(\varepsilon) \rceil$, we know $r \ge 3/p_{\mathsf{rej}}(\varepsilon)$, thus

$$\Pr[F] \le \left(1 - \frac{11 p_{\mathsf{rej}}(\varepsilon)}{12}\right)^{3/p_{\mathsf{rej}}(\varepsilon)} \le e^{\frac{-11 p_{\mathsf{rej}}(\varepsilon)}{12} \cdot \frac{3}{p_{\mathsf{rej}}(\varepsilon)}} = \left(\frac{1}{e}\right)^{11/4} \le \frac{1}{3},$$

where the second inequality holds since $1 - x \le e^{-x}$ for all x. Thus, the failure probability of $\mathcal{A}$ is at most $1/3$, and $\mathcal{A}$ rejects f with probability at least $2/3$. ◀

5 Resilience to online manipulations for general groups

In this section, we prove Theorem 1.1 by lifting the linearity tester in [4], which is based on the manipulation-resilient tester of [10], and the sample-based tester of Goldreich and Ron [25]. Define a parameter $m := 4\lceil \log_2 t + 15/\varepsilon \rceil + 12$. Depending on the value of m, we invoke one of the two different algorithms against the t-online-erasure adversary. Following the case-based strategy of [4], when $2^m \le |G|^{1/4}$ we use Algorithm 4, a lift of the linearity tester of [10] to the general group setting. On the other hand, when $2^m > |G|^{1/4}$, we instead use the sample-based tester by [25]. We analyze the two cases separately. Our main contribution here is in the analysis of the distribution of the element y in Algorithm 3, where we need to handle random signs along with random elements for any group G, whereas previous work did not have to worry about the random signs.

Case 1: $2^m \le |G|^{1/4}$. In this case, we use Algorithm 4 which internally runs Algorithm 3 a constant number of times.

Algorithm 3 Unpredictable Random Signs Test.

Parameters: even $m \in \mathbb{N}$
Input: Query access to $f : G \to H$ where $(G, +)$ and $(H, \oplus)$ are finite groups.
1: Draw m independent and uniformly random elements $x_1, x_2, \ldots, x_m$ from G.
2: Query $f(x_1), f(x_2), \ldots, f(x_m)$.
3: Draw independent and uniformly random signs σ_j for all $j \in [m]$ from the set $\{+, -\}$.
4: Let S be a uniformly random subset of $[m]$ of size $m/2$.
5: Query $f(y)$ where $y \leftarrow \sum_{j \in S} \sigma_j x_j$
6: **Reject** if $\bigoplus_{j \in S} \sigma_j f(x_j) \ne f(y)$. ▷ This implies no erasures were encountered.
7: **Accept**.

Note that Algorithm 3 has the same completeness and soundness as $\mathsf{Random\ Signs\ Test}_{m/2}$. Moreover, Algorithm 3 is also unpredictable as stated in Lemma 5.1 whose proof we defer to the full version.

▶ **Lemma 5.1.** *Fix a finite group G. If $2^m \le |G|^{1/4}$, then Algorithm 3 with parameter m is*

$$\left(m + 1, \frac{m}{2^m} + \frac{1}{\binom{m}{m/2}}, \frac{1}{2^m}\right)\text{-unpredictable.}$$

Algorithm 4 Online Erasure-Resilient Random Signs Test.

Parameters: $\varepsilon \in (0,1)$, $t \in \mathbb{N}$.
Input: Query access to $f : G \to H$ via t-erasure oracle, where $(G,+)$ and $(H,\oplus)$ are finite groups.
1: Set $m = 4\lceil \log_2 t + 15/\varepsilon \rceil + 12$
2: **for** $i \in [48]$ **do**
3: Run Algorithm 3 with parameter m, independent randomness and query access to f.
4: **Reject** if Algorithm 3 rejects.
5: **Accept**.

Proof of Theorem 1.1 when $2^m \leq |G|^{1/4}$. Finally, we prove that Algorithm 4 satisfies Theorem 1.1 by an application of the general amplification for online resilience lemma (Lemma 4.4). The details are deferred to the full version. ◀

Case 2: $2^m > |G|^{1/4}$. In this case, we use the sample-based tester by Goldreich and Ron [25] and use a theorem by Arora, Kelman and Meir [4] about online-manipulation-resilience of sample-based testers to handle this case. The details are deferred to the full version.

References

1 Noga Alon, Tali Kaufman, Michael Krivelevich, Simon Litsyn, and Dana Ron. Testing Reed-Muller codes. *IEEE Transactions on Information Theory*, 51(11):4032–4039, 2005. doi:10.1109/TIT.2005.856958.

2 Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998. doi:10.1145/278298.278306.

3 Sanjeev Arora and Madhu Sudan. Improved low-degree testing and its applications. *Combinatorica*, 23(3):365–426, 2003. doi:10.1007/s00493-003-0025-0.

4 Vipul Arora, Esty Kelman, and Uri Meir. On optimal testing of linearity. In Ioana Oriana Bercea and Rasmus Pagh, editors, *2025 Symposium on Simplicity in Algorithms, SOSA 2025, New Orleans, LA, USA, January 13-15, 2025*, pages 65–76. SIAM, 2025. doi:10.1137/1.9781611978315.5.

5 László Babai, Lance Fortnow, Leonid A. Levin, and Mario Szegedy. Checking computations in polylogarithmic time. In *STOC*, pages 21–31, 1991. doi:10.1145/103418.103428.

6 László Babai, Lance Fortnow, and Carsten Lund. Non-deterministic exponential time has two-prover interactive protocols. *Computational Complexity*, 1:3–40, 1991. doi:10.1007/BF01200056.

7 Oren Becker and Michael Chapman. Stability of approximate group actions: uniform and probabilistic. *Journal of the European Mathematical Society*, 25(9):3599–3632, 2022.

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. doi:10.1109/18.556674.

9 Aleksandrs Belovs. Adaptive lower bound for testing monotonicity on the line. In *Proceedings of Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM)*, pages 31:1–31:10, 2018. doi:10.4230/LIPIcs.APPROX-RANDOM.2018.31.

10 Omri Ben-Eliezer, Esty Kelman, Uri Meir, and Sofya Raskhodnikova. Property testing with online adversaries. *ACM Transactions on Computation Theory*, 18(1):1–40, 2026. doi:10.1145/3778165.

11 Michael Ben-Or, Don Coppersmith, Michael Luby, and Ronitt Rubinfeld. Non-abelian homomorphism testing, and distributions close to their self-convolutions. *Random Struct. Algorithms*, 32(1):49–70, 2008. doi:10.1002/RSA.20182.

12 Eli Ben-Sasson, Madhu Sudan, Salil P. Vadhan, and Avi Wigderson. Randomness-efficient low degree tests and short PCPs via epsilon-biased sets. In *STOC*, pages 612–621, 2003. doi:10.1145/780542.780631.

13 Arnab Bhattacharyya, Elena Grigorescu, Kyomin Jung, Sofya Raskhodnikova, and David P. Woodruff. Transitive-closure spanners. *SIAM Journal on Computing*, 41(6):1380–1425, 2012. doi:10.1137/110826655.

14 Arnab Bhattacharyya, Swastik Kopparty, Grant Schoenebeck, Madhu Sudan, and David Zuckerman. Optimal testing of Reed-Muller codes. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, Las Vegas, Nevada, USA, October 23-26, 2010*, pages 488–497, 2010. doi:10.1109/FOCS.2010.54.

15 Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3):549–595, 1993. doi:10.1016/0022-0000(93)90044-W.

16 Deeparnab Chakrabarty and C. Seshadhri. Optimal bounds for monotonicity and Lipschitz testing over hypercubes and hypergrids. In *STOC*, pages 419–428, 2013. doi:10.1145/2488608.2488661.

17 Irit Dinur and Venkatesan Guruswami. PCPs via low-degree long code and hardness for constrained hypergraph coloring. *Israel Journal of Mathematics*, 209(2):611–649, 2015. doi:10.1007/s11856-015-1231-3.

18 John D Dixon. The probability of generating the symmetric group. *Mathematische Zeitschrift*, 110:199–205, 1969.

19 Yevgeniy Dodis, Oded Goldreich, Eric Lehman, Sofya Raskhodnikova, Dana Ron, and Alex Samorodnitsky. Improved testing algorithms for monotonicity. In *Proceedings of Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM)*, pages 97–108, 1999. doi:10.1007/978-3-540-48413-4_10.

20 Funda Ergün, Sampath Kannan, Ravi Kumar, Ronitt Rubinfeld, and Mahesh Viswanathan. Spot-checkers. *Journal of Computer and System Sciences*, 60(3):717–751, 2000. doi:10.1006/jcss.1999.1692.

21 Uriel Feige, Shafi Goldwasser, László Lovász, Shmuel Safra, and Mario Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43(2):268–292, 1996. doi:10.1145/226643.226652.

22 Katalin Friedl and Madhu Sudan. Some improvements to total degree tests. In *Third Israel Symposium on Theory of Computing and Systems (ISTCS)*, pages 190–198, 1995. doi:10.1109/ISTCS.1995.377032.

23 Peter Gemmell, Richard J. Lipton, Ronitt Rubinfeld, Madhu Sudan, and Avi Wigderson. Self-testing/correcting for polynomials and for approximate functions. In *STOC*, pages 32–42, 1991. doi:10.1145/103418.103429.

24 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45(4):653–750, 1998. doi:10.1145/285055.285060.

25 Oded Goldreich and Dana Ron. On sample-based testers. *ACM Trans. Comput. Theory*, 8(2):7:1–7:54, 2016. doi:10.1145/2898355.

26 William Timothy Gowers and Omid Hatami Varzaneh. Inverse and stability theorems for approximate representations of finite groups. *Matematicheskii Sbornik*, 208(12):70–106, 2017.

27 Elena Grigorescu, Swastik Kopparty, and Madhu Sudan. Local decoding and testing for homomorphisms. In Josep Díaz, Klaus Jansen, José D. P. Rolim, and Uri Zwick, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 9th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2006 and 10th International Workshop on Randomization and Computation, RANDOM 2006, Barcelona, Spain, August 28-30 2006, Proceedings*, volume 4110 of *Lecture Notes in Computer Science*, pages 375–385. Springer, 2006. doi:10.1007/11830924_35.

28 Elad Haramaty, Amir Shpilka, and Madhu Sudan. Optimal testing of multivariate polynomials over small prime fields. *SIAM Journal on Computing*, 42(2):536–562, 2013. doi:10.1137/120879257.

29 Johan Håstad and Avi Wigderson. Simple analysis of graph tests for linearity and PCP. *Random Struct. Algorithms*, 22(2):139–160, 2003. doi:10.1002/rsa.10068.

30 Charanjit S. Jutla, Anindya C. Patthak, Atri Rudra, and David Zuckerman. Testing low-degree polynomials over prime fields. *rsa*, 35(2):163–193, 2009. doi:10.1002/rsa.20262.

31 Iden Kalemaj, Sofya Raskhodnikova, and Nithin Varma. Sublinear-time computation in the presence of online erasures. *Theory of Computing*, 19(1):1–48, 2023. doi:10.4086/TOC.2023.V019A001.

32 Tali Kaufman and Dor Minzer. Improved optimal testing results from global hypercontractivity. *SIAM Journal on Computing*, 54(3):625–663, 2025. doi:10.1137/23M1571022.

33 Tali Kaufman and Dana Ron. Testing polynomials over general fields. *SIAM Journal on Computing*, 36(3):779–802, 2006. doi:10.1137/S0097539704445615.

34 Esty Kelman, Ephraim Linder, and Sofya Raskhodnikova. Online versus offline adversaries in property testing. In Raghu Meka, editor, *16th Innovations in Theoretical Computer Science Conference, ITCS 2025, January 7-10, 2025, Columbia University, New York, NY, USA*, volume 325 of *LIPIcs*, pages 65:1–65:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.ITCS.2025.65.

35 Esty Kelman, Uri Meir, Debanuj Nayak, and Sofya Raskhodnikova. Homomorphism testing with resilience to online manipulations. *CoRR*, abs/2511.23363, 2025. doi:10.48550/arXiv.2511.23363.

36 Esty Kelman, Uri Meir, and Kai Zhe Zheng. Optimal testing of reed-muller codes with an online adversary. In Dana Moshkovitz, editor, *41st Computational Complexity Conference (CCC 2026)*, volume 383 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 31:1–31:26, Dagstuhl, Germany, 2026. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.CCC.2026.31.

37 Subhash Khot and Kunal Mittal. Biased Linearity Testing in the 1% Regime. In Srikanth Srinivasan, editor, *40th Computational Complexity Conference (CCC 2025)*, volume 339 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 10:1–10:23, Dagstuhl, Germany, 2025. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.CCC.2025.10.

38 Martin W Liebeck and Aner Shalev. The probability of generating a finite simple group. *Geometriae dedicata*, 56:103–113, 1995.

39 Alexander Lubotzky. The expected number of random elements to generate a finite group. *Journal of Algebra*, 257(2):452–459, 2002.

40 Andrea Lucchini. The expected number of random elements to generate a finite group. *Monatshefte für Mathematik*, 181(1):123–142, 2016.

41 Nina E Menezes. *Random generation and chief length of finite groups*. PhD thesis, University of St Andrews, 2013.

42 Dor Minzer and Kai Zhe Zheng. Adversarial low degree testing. In David P. Woodruff, editor, *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024, Alexandria, VA, USA, January 7-10, 2024*, pages 4395–4409. SIAM, 2024. doi:10.1137/1.9781611977912.154.

43 Tushant Mittal and Sourya Roy. Derandomized non-abelian homomorphism testing in low soundness regime. *arXiv preprint arXiv:2405.18998*, 2024. doi:10.48550/arXiv.2405.18998.

44 Tushant Mittal and Sourya Roy. A general framework for low soundness homomorphism testing. In Shubhangi Saraf, editor, *17th Innovations in Theoretical Computer Science Conference (ITCS 2026)*, volume 362 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 103:1–103:18, Dagstuhl, Germany, 2026. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ITCS.2026.103.

45 Cristopher Moore and Alexander Russell. Approximate representations, approximate homomorphisms, and low-dimensional embeddings of groups. *SIAM Journal on Discrete Mathematics*, 29(1):182–197, 2015. doi:10.1137/140958578.

46 Dana Moshkovitz and Ran Raz. Sub-constant error low degree test of almost-linear size. *SIAM Journal on Computing*, 38(1):140–180, 2008. doi:10.1137/060656838.

47 Carl Pomerance. The expected number of random elements to generate a finite abelian group. *Periodica Mathematica Hungarica*, 43(1):191–198, 2002. doi:10.1023/A\%3A1015250102792.

48 Sofya Raskhodnikova. Monotonicity testing. *Masters Thesis, MIT*, 1999.

49 Sofya Raskhodnikova. Testing if an array is sorted. In *Encyclopedia of Algorithms*, pages 2219–2222. Springer, New York, NY, 2016. doi:10.1007/978-1-4939-2864-4_700.

50 Sofya Raskhodnikova and Ronitt Rubinfeld. Linearity testing/testing Hadamard codes. In *Encyclopedia of Algorithms*, pages 1107–1110. Springer, 2016. doi:10.1007/978-1-4939-2864-4_202.

51 Ran Raz and Shmuel Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *STOC*, pages 475–484, 1997. doi:10.1145/258533.258641.

52 Noga Ron-Zewi and Madhu Sudan. A new upper bound on the query complexity of testing generalized Reed-Muller codes. *Theory Comput.*, 9:783–807, 2013. doi:10.4086/toc.2013.v009a025.

53 Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996. doi:10.1137/S0097539793255151.

54 Alex Samorodnitsky. Low-degree tests at large distances. In *STOC*, pages 506–515, 2007. doi:10.1145/1250790.1250864.

55 Alex Samorodnitsky and Luca Trevisan. A PCP characterization of NP with optimal amortized query complexity. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA*, pages 191–199, 2000. doi:10.1145/335305.335329.

56 Alex Samorodnitsky and Luca Trevisan. Gowers uniformity, influence of variables, and PCPs. *SIAM Journal on Computing*, 39(1):323–360, 2009. doi:10.1137/070681612.

57 Miklos Santha and Umesh V Vazirani. Generating quasi-random sequences from semi-random sources. *Journal of computer and system sciences*, 33(1):75–87, 1986. doi:10.1016/0022-0000(86)90044-9.

58 Amir Shpilka and Avi Wigderson. Derandomizing homomorphism testing in general groups. *SIAM Journal on Computing*, 36(4):1215–1230, 2006. doi:10.1137/S009753970444658X.

59 Luca Trevisan. Recycling queries in PCPs and in linearity tests (extended abstract). In *STOC*, pages 299–308, 1998. doi:10.1145/276698.276769.

60 Alek Westover, Edward Yu, and Kai Zhe Zheng. New Direct Sum Tests. In *16th Innovations in Theoretical Computer Science Conference (ITCS 2025)*, volume 325 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 94:1–94:26. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.ITCS.2025.94.