Electronic Colloquium on Computational Complexity, Report No. 123 (2018)

Probabilistic Checking against Non-Signaling Strategies from Linearity Testing

Alessandro Chiesa
alexch@berkeley.edu
UC Berkeley

Peter Manohar
manohar@berkeley.edu
UC Berkeley

Igor Shinkar
igors@berkeley.edu
UC Berkeley

Abstract

Non-signaling strategies are a generalization of quantum strategies that have been studied in physics over the past three decades. Recently, they have found applications in theoretical computer science, including to proving inapproximability results for linear programming and to constructing protocols for delegating computation. A central tool for these applications is probabilistically checkable proofs (PCPs) that are *sound against non-signaling strategies*.

In this paper we prove that the exponential-length constant-query PCP construction due to Arora et al. (JACM 1998) is sound against non-signaling strategies.

Our result offers a new length-vs-query tradeoff when compared to the non-signaling PCP of Kalai et al. (STOC 2013 and 2014) and, moreover, provides additional evidence to the conjecture that a non-signaling analogue of the PCP Theorem holds.

Keywords: probabilistically checkable proofs; linearity testing; non-signaling strategies

ISSN 1433-8092

Contents

1 Introduction

Probabilistically Checkable Proofs (PCPs) [BFLS91; FGLSS96; AS98; ALMSS98] are proofs whose validity can be checked by a probabilistic verifier that accesses only a few locations of the proof. PCPs have numerous applications across the theory of computing, including to hardness of approximation [FGLSS96] and delegation of computation [Kil92; Mic00].

A seminal result, known as the PCP Theorem [AS98; ALMSS98], states that every language in $\mathsf{NTIME}(T)$ can be probabilistically checked by a verifier that uses $O(\log T)$ random bits and makes $O(1)$ queries to a proof of length $\mathsf{poly}(T)$.[1]

In this paper we study PCPs that are sound against *non-signaling strategies* (nsPCPs). These have recently found applications that appear out of the reach of (standard) PCPs, including 1-round delegation of computation from falsifiable assumptions [KRR13; KRR14] and hardness of approximation for linear programming [KRR16]. The efficiency measures achieved in known nsPCPs appear suboptimal, which affects the quality of the corresponding applications. We thus ask whether a non-signaling analogue of the PCP Theorem holds.

Below we explain the aforementioned notions, and then present our results in this direction.

Non-signaling strategies. Non-signaling strategies are a class of "non-local" correlations that strictly generalize quantum strategies, and capture the minimal condition that spatially-isolated parties cannot communicate instantaneously. They have been studied in physics for over three decades [Ras85; KT92; PR94] in order to better understand quantum entanglement.

There are two definitions, corresponding to whether the strategy is meant to represent a function or isolated parties; the former is the relevant one for nsPCPs [KRR13; KRR14].[2] Given a locality parameter $k \in \mathbb{N}$, a *k-non-signaling function* $\mathcal{F}$ extends the notion of a function $f \colon D \to \{0,1\}$ as follows: it is a collection $\{\mathcal{F}_S\}_{S \subseteq D, |S| \leq k}$ where each $\mathcal{F}_S$ is a *distribution* over $\{0,1\}^S$ and, for every two subsets S_1 and S_2 each of size at most k, the restrictions of $\mathcal{F}_{S_1}$ and $\mathcal{F}_{S_2}$ to $S_1 \cap S_2$ are equal as distributions. Note that if $k = |D|$ then $\mathcal{F}$ is a distribution over functions $f \colon D \to \{0,1\}$.

Note that k-non-signaling functions are solutions to the linear program arising from the k-relaxation in the Sherali–Adams hierarchy [SA90]. The variables are of the form $X_{S,\vec{b}}$ (for all $S \subseteq D$ of size at most k and $\vec{b} \in \{0,1\}^S$) and express the probability of $\vec{b}$ in the distribution $\mathcal{F}_S$; consistency across subsets S and T is expressed using the natural linear constraints.[3]

Non-signaling PCPs. Recall that a classical PCP verifier is given oracle access to a proof represented as a function $f \colon D \to \{0,1\}$. The verifier uses random bits, makes a few queries to f, and then accepts or rejects. Completeness requires that if the statement being checked is true then there is a function f that makes the verifier always accept. Soundness requires that if the statement being checked is false then every function f makes the verifier reject with high probability.

In the non-signaling setting, "proofs" are non-signaling functions rather than (classical) functions. Soundness is correspondingly stronger: given a locality parameter $k \in \mathbb{N}$, soundness requires that every *k-non-signaling function* $\mathcal{F}$ makes the nsPCP verifier reject with high probability.

Efficiency measures of a nsPCP include familiar notions such as proof length (defined as $|D|$) and the verifier's randomness and query complexity. In addition, the locality parameter k controls

[1] In particular, for every language in $\mathsf{NEXP} = \cup_{c \in \mathbb{N}} \mathsf{NTIME}(2^{n^c})$, the verifier uses $\mathsf{poly}(n)$ random bits and makes $O(1)$ queries to a proof of length $2^{\mathsf{poly}(n)}$.

[2] The other definition underlies the notion of multi-prover interactive proofs that are sound against non-signaling strategies (nsMIPs). Any nsPCP gives rise to an nsMIP with similar parameters. See [KRR13; KRR14] for details.

[3] In fact it suffices to only have variables of the form $X_{S,1^S}$ since all other probabilities can be computed from these.

how hard it is to attain soundness: the smaller k is, the larger the set of non-signaling functions that the verifier could face. (Note that k-non-signaling implies $(k-1)$-non-signaling.)

There is a qualitative difference between the complexity classes captured by PCPs and by nsPCPs; namely, while PCPs capture *non-deterministic* time languages, nsPCPs capture *deterministic* ones. Indeed, the aforementioned PCP Theorem implies that it is NEXP-hard to approximate the maximum acceptance probability of a PCP verifier. In contrast, computing the maximum acceptance probability of an nsPCP verifier that uses r random bits reduces to a linear program with $2^{\mathsf{poly}(rk)}$ variables and constraints, a problem solvable in $\mathsf{EXP} = \cup_{c \in \mathbb{N}} \mathsf{DTIME}(2^{n^c})$.

If $k = 2$, the linear program is solvable in PSPACE [Ito10], which is a tight upper bound [IKM09]. For $k > 2$ little is known, except for a seminal result of Kalai, Raz, and Rothblum [KRR13; KRR14], which shows that for $k = \mathsf{poly}(n)$ it is EXP-hard to approximate a nsPCP verifier's maximum acceptance probability. In more detail, every language in $\mathsf{DTIME}(T)$ has a verifier that uses $\mathsf{poly}(\log T)$ random bits and makes $\mathsf{poly}(\log T)$ queries to a proof of length $\mathsf{poly}(T)$; soundness holds against $\mathsf{poly}(\log T)$-non-signaling functions; the verifier runs in time $n \cdot \mathsf{poly}(\log T)$ and space $\mathsf{poly}(\log T)$.[4]

The nsPCP Conjecture. The nsPCP construction behind the above result is a whitebox modification of early PCP constructions [BFL91; BFLS91], and achieves efficiency similar to those. However, modern "PCP technology" goes well beyond these early constructions, via tools such as proof composition [AS98] and proofs of proximity [DR04; BGHSV06], and enables better efficiency, including the PCP Theorem. Yet, current "nsPCP technology" is limited to the above results, and the question of whether a non-signaling analogue of the PCP Theorem holds remains open.

Question 1.1. *Is it true that every language in* $\mathsf{DTIME}(T)$ *has an nsPCP verifier that uses* $O(\log T)$ *random bits, makes* $O(1)$ *queries, and is sound against* $O(1)$*-non-signaling functions?*
(As above, we also require that the verifier runs in time $n \cdot \mathsf{poly}(\log T)$ *and space* $\mathsf{poly}(\log T)$*.)*

An affirmative answer to the above question would, e.g., improve the hardness result for linear programming in [KRR16], by yielding a reduction that outputs a linear program of polynomial, rather than a quasipolynomial, size. While we do not know if an affirmative answer exists (and we cannot prove that it does not exist), it is clear that the (very few) tools that we have to construct and analyze nsPCPs are far from this goal. In this paper we make headway towards this goal.

1.1 Towards a nsPCP Theorem

In [ALMSS98] a key step towards the PCP Theorem is to prove a weaker result in which the proof has *exponential*, rather than *polynomial*, size (and so the randomness complexity of the verifier is polynomial rather than logarithmic). Namely, one proves that every language in $\mathsf{NTIME}(T)$ has a PCP verifier that uses $\mathsf{poly}(T)$ random bits and makes $O(1)$ queries to a proof of length $2^{\mathsf{poly}(T)}$.

In this paper we ask whether a non-signaling analogue of this result holds for the class $\mathsf{DTIME}(T)$.

Question 1.2. *Is it true that every language in* $\mathsf{DTIME}(T)$ *has an nsPCP verifier that uses* $\mathsf{poly}(T)$ *random bits, makes* $O(1)$ *queries, and is sound against* $O(1)$*-non-signaling functions?*

We propose this question as a relaxation that, not only is interesting in its own right, but is likely to shed light on Question 1.1. However, one must be careful with the precise formulation of

[4] Achieving time and space complexities that are $o(T)$ is important for applications. This is not surprising as every language in $\mathsf{DTIME}(T)$ has a trivial nsPCP verifier that runs in time T: the verifier that simply decides the language, without asking any queries. This is unlike the case of PCPs for $\mathsf{NTIME}(T)$, where time complexity is less critical.

Question 1.2. If the verifier can use $\mathsf{poly}(T)$ random bits then it can simply decide the language by running in time T, without making any queries. To recover a meaningful question, we *require* that in order to decide whether an instance x is in a language $L \in \mathsf{DTIME}(T)$ the nsPCP verifier first generates queries via a $\mathsf{poly}(T)$-time sampler that is *input oblivious* (knows the length of x but not x itself), and then rules according to a $o(T)$-time decision predicate that knows x. We stress that all PCP/nsPCP verifiers discussed in this paper are input oblivious.

In this paper we study Question 1.2 by analyzing a natural candidate construction, and ask:

Is the exponential-length $O(1)$-query PCP of [ALMSS98] sound against $O(1)$-non-signaling functions?

Hereafter, we consider the complexity class $\mathsf{DSIZE}(S)$ (languages decidable by uniform circuits of size $S(n)$) instead of the class $\mathsf{DTIME}(T)$ (languages decidable by machines in time $T(n)$) because our results, like their classical counterparts, are most easily stated in terms of uniform circuits. This change is only for simplicity, as $\mathsf{DTIME}(T) \subseteq \mathsf{DSIZE}(\mathsf{poly}(T))$.

1.2 Main theorem

In this paper we prove that the exponential-length constant-query PCP construction of [ALMSS98] (without modifications) is sound against non-signaling functions. We obtain the following theorem.

Theorem 1 (main theorem)**.** *Every language $L \in \mathsf{DSIZE}(S)$ has an input-oblivious nsPCP verifier that uses $O(S^2)$ random bits, makes 11 queries, and is sound against $O(\log^2 S)$-non-signaling functions. The query sampler runs in time $O(S^2)$, and the decision predicate runs in time $O(n)$.*

The theorem is close to answering Question 1.2, which asks for soundness against $O(1)$-non-signaling functions. (See Fig. 1 for a comparison with the classical result on nondeterministic languages.) At the same time, some may consider Ito's algorithm [Ito10] as evidence that soundness against $O(1)$-non-signaling functions is too much to hope for. Understanding this gap needs further research.

Our result is *incomparable* to the nsPCP of [KRR13; KRR14], where the nsPCP verifier uses $\mathsf{poly}(\log S)$ random bits to make $\mathsf{poly}(\log S)$ queries. The fact we prove soundness against $O(\log^2 S)$-non-signaling functions is somewhat undesirable, as this implies that the corresponding nsMIP requires $O(\log^2 S)$ provers. That said, the nsMIP of [KRR13; KRR14] requires many more provers: $\mathsf{poly}(\log S)$ with the degree in the polynomial much larger than 2. Another feature of our nsMIP is that we have "room" to repeat the verifier $O(\log^2 S)$ times without requiring any additional provers, achieving soundness error $2^{-O(\log^2 S)}$ against $O(\log^2 S)$ provers. This is not a real drawback in cryptographic applications because one needs to anyhow reduce soundness error.

Finally, our result is the first to demonstrate that a classical PCP construction is secure against non-signaling functions, *without any modifications*. This should be compared to the construction considered in [KRR13; KRR14] that, while modeled after the PCP in [BFL91; BFLS91], includes several notable modifications that are needed in the soundness proof.

1.3 Main lemmas

We outline the ideas behind our theorem in Section 2. Concretely, we highlight several statements, which we deem of independent interest, that we prove on the way to the theorem.

Recall that the exponential-length constant-query PCP in [ALMSS98] is obtained in two steps. First, construct a constant-query verifier where soundness holds as long as the proof string is a *linear*

		complexity	type of	soundness	proof			
construction	reference	class	PCP	error	length	randomness	queries	locality
ALMSS verifier	[ALMSS98]	$\mathsf{NSIZE}(S)$	PCP	$1-1/36$	$2^{O(S^2)}$	$O(S^2)$	11	n/a
+ linearity test	Theorem 1	$\mathsf{DSIZE}(S)$	ns PCP	$1-1/10^5$				$O(\log^2 S)$
ALMSS verifier	[ALMSS98]	$\mathsf{NSIZE}(S)$	LPCP	$3/4$	$O(S^2)$	$O(S)$	4	n/a
	Theorem 2	$\mathsf{DSIZE}(S)$	ns LPCP	$5/6$				$O(\log S)$

Figure 1: The (linear) ALMSS verifier in different PCP settings.

function; this is known as a *linear PCP*. Second, use a linearity test [BLR93] and self-correction to *compile* this linear PCP into a (standard) PCP, where soundness holds against arbitrary proofs.

Our approach follows the same two steps, but adapted to the non-signaling setting. This also departs from the approach in [KRR14], which does not make use of any property testing results.

Note, however, that it is a priori not clear what is the non-signaling analogue of a linear function. A natural attempt would be to say that a non-signaling function $\mathcal{F}$ is linear iff it passes the BLR linearity test with probability 1 (where the probability is over the test and $\mathcal{F}$). But this attempt is awkward, because the definition depends on a local test, and avoids discussing "global" structure.

A recent work [CMS18] tells us that the right definition is to say that $\mathcal{F}$ is linear iff it corresponds to a *quasi-distribution* over linear functions. A quasi-distribution is a probability distribution where the weights can be any real number and are not restricted to be in $[0, 1]$. Quasi-distributions over functions arise in this context because they are an equivalent description of non-signaling functions.

In light of the above, the notion of a *non-signaling linear PCP* (nsLPCP) is immediate: the definition requires soundness to hold against all *linear* non-signaling functions.

The first step in our proof is showing that the linear PCP verifier of [ALMSS98] (the "ALMSS verifier"), when used for deterministic computations, is sound against linear non-signaling functions.

Theorem 2. *The (input oblivious) ALMSS verifier, for a given language $L \in \mathsf{DSIZE}(S)$, uses $O(S)$ random bits, makes 4 queries, and is sound against linear $O(\log S)$-non-signaling functions.*

See Fig. 1 for a comparison with the classical result showing soundness against linear functions.

In order to "lift" Theorem 2 to Theorem 1, we need a suitable linearity test.

The linearity test of [BLR93] was recently analyzed in the non-signaling setting by [CMS18], who proved that any k-non-signaling function $\mathcal{F}$ that passes the linearity test with probability $1-\varepsilon$ can be self-corrected to a $\lfloor k/2 \rfloor$-non-signaling function $\hat{\mathcal{F}}$ that is $2^{O(k)}\varepsilon$-close to a linear $\lfloor k/2 \rfloor$-non-signaling function $\mathcal{L}$. (Self-correction and closeness have precise meanings, discussed later.) However, we cannot directly use [CMS18]'s result, because in our theorem the locality parameter k is required to be super-constant ($k = O(\log S)$ in Theorem 2), and thus the bound on the distance between $\hat{\mathcal{F}}$ and $\mathcal{L}$ is too large, even when considering only query sets of small size. Specifically, we need the distance to be a sufficiently small constant on query sets of size 4 (the number of queries in Theorem 2).

We solve this problem by extending the result in [CMS18] in a black-box way and proving that the distance between $\hat{\mathcal{F}}$ and $\mathcal{L}$ on a query set Q is only $O(|Q|\sqrt{\varepsilon})$, provided that the error ε and $\mathcal{L}$'s locality are sufficiently small. Crucially, if $|Q|$ is constant, so is the distance between $\hat{\mathcal{F}}$ and $\mathcal{L}$. The proof of this statement involves analyzing the *repeated* linearity test, whose behavior in the non-signaling setting is quite subtle when compared to the classical setting (see Section 2.6).

Theorem 3. *Let $k, \bar{k} \in \mathbb{N}$ and $\varepsilon \in (0, 1/400]$ be such that $k \geq \Omega((\bar{k} + \log \frac{1}{\varepsilon}) \cdot \bar{k})$. If a k-non-signaling function $\mathcal{F} \colon \{0,1\}^n \to \{0,1\}$ passes the linearity test with probability at least $1 - \varepsilon$ then there exists a linear $\bar{k}$-non-signaling function $\mathcal{L} \colon \{0,1\}^n \to \{0,1\}$ such that for all query sets $Q \subseteq \{0,1\}^n$ with size $|Q| \leq \bar{k}$ and for all events $E \subseteq \{0,1\}^Q$ it holds that*

$$\left| \Pr[\hat{\mathcal{F}}(Q) \in E] - \Pr[\mathcal{L}(Q) \in E] \right| \leq O(|Q| \sqrt{\varepsilon}) \ .$$

The above result on linearity testing enables us to transform our nsLPCP, and more generally *any* nsLPCP, into a corresponding nsPCP with minimal changes in parameters (the transformation is exactly the classical compiler). This is the last key statement in the proof of our main theorem.

Lemma 1.3. *For every $\varepsilon \in [0, 1]$, if a language L has an nsLPCP where the verifier uses r random bits, makes q queries, and has soundness error $1 - \varepsilon$ against linear k-non-signaling functions $\mathcal{L} \colon \{0,1\}^\ell \to \{0,1\}$, then L has an nsPCP where the verifier uses $r + O(q\ell)$ random bits, makes $O(q)$ queries, and has soundness error $1 - O_q(\varepsilon^2)$ against $O_\varepsilon(k^2)$-non-signaling functions $\mathcal{F} \colon \{0,1\}^\ell \to \{0,1\}$. (Furthermore, if the former is input oblivious, so is the latter.)*

1.4 Enriching the toolkit for non-signaling PCPs

Progress in our understanding of PCPs has typically moved hand in hand with progress in our understanding of low-degree testing. In particular, many PCP constructions follow this blueprint: (1) a low-degree test that, via only a few queries, ensures that a given proof conforms to a specified algebraic encoding; (2) a probabilistic test that, assuming the proof is (essentially) given in this encoding, ensures that the statement being checked is true with high probability.

In contrast, while the nsPCP in [KRR14] is reminiscent of this blueprint, its analysis does not follow it, despite the fact that the construction is modeled after the PCP in [BFL91; BFLS91], for which the two-step analysis *is* possible (in the classical setting). The lack of such general paradigms means that we lack general design principles to construct better nsPCPs.

This state of affairs raises the intriguing question of whether a theory of low-degree testing (and, more generally, property testing) is feasible in the non-signaling setting and, moreover, whether one can build on it to construct nsPCPs in order to make further progress towards Question 1.1.

An additional contribution of our work is to *enrich* the current "non-signaling toolkit", by demonstrating an example where the aforementioned blueprint is both possible and useful.

Namely, building on the work of [CMS18] on linearity testing, our results provide a modular paradigm that not only simplifies the overall analysis, thereby enabling us to assert that the construction of [ALMSS98] *with no modifications* is sound against non-signaling strategies, but also (as discussed later) clarifies the technical barriers that separate us from answering Question 1.2. All this suggests that our techniques will be helpful for constructing more efficient nsPCPs.

1.5 Open problems

The question of whether the exponential-length constant-query PCP of [ALMSS98] is sound against $O(1)$-non-signaling functions remains open. A concrete approach to affirmatively answer this question is to prove that the *linear* PCP verifier of [ALMSS98] is sound against k-non-signaling functions for $k = O(1)$, rather than $k = O(\log S)$ as in Theorem 2. (Our generic compiler from Lemma 1.3 would then take care of the rest.) Another intriguing possibility is that an affirmative answer to Question 1.2 could come from a *different* exponential-size constant-query PCP. However, the result due to [Ito10] shows that the class of nsMIPs with 2 provers equals PSPACE, which possibly suggests that soundness against $O(1)$-non-signaling functions is too much to hope for.

Moreover, while our results can be interpreted as progress towards a non-signaling analogue of the PCP Theorem (Question 1.1), it remains unclear whether such an analogue holds, and more investigations in nsPCPs are needed. We believe that our work and our new techniques can inform such investigations.

2 Techniques

We outline the techniques used to prove our results. First, in Section 2.1, we explain the transformation from a nsLPCP to a corresponding nsPCP. Next, in Sections 2.2 to 2.5 we discuss the nsLPCP on which we apply this transformation, namely, the ALMSS verifier [ALMSS98]. Finally, in Section 2.6, we discuss linearity testing with low error, which underlies the transformation.

2.1 From nsLPCP to nsPCP

We discuss the transformation from nsLPCP to nsPCP (Lemma 1.3). We first recall the classical transformation from LPCP to PCP, and then explain how to achieve its non-signaling analogue.

The classical case. The classical transformation from LPCP to PCP relies on the following tools.

- *Testing linearity.* Given a boolean function $f\colon \{0,1\}^\ell \to \{0,1\}$, the linearity test draws random $x, y \in \{0,1\}^\ell$ and checks that $f(x) + f(y) = f(x+y)$ [BLR93]. If the test passes with probability $1-\varepsilon$, then f is ε-close to a linear function $f^*\colon \{0,1\}^\ell \to \{0,1\}$ [BLR93; BCHKS96].

- *Self-correction.* Given f that is ε-close to a linear function f^*, one can create a probabilistic oracle $\mathcal{O}$ that, given *any* $x \in \{0,1\}^\ell$, returns $f^*(x)$ with probability $1-2\varepsilon$. Namely, $\mathcal{O}$ samples a random $z \in \{0,1\}^\ell$, queries f on $z+x$ and z, and answers with $f(z+x) - f(z)$.

The above tools imply a transformation from LPCP to PCP: given access to an arbitrary function $f\colon \{0,1\}^\ell \to \{0,1\}$, the PCP verifier runs the linearity test and then runs the LPCP verifier by self-correcting each of its queries. If the LPCP verifier makes q queries and has soundness error γ, then the resulting PCP verifier makes $3+2q$ queries and has soundness error $\max\{1-\varepsilon, \gamma+2q\varepsilon\}$, where ε is (a bound on) the distance of f to linear functions. This soundness error is bounded by $1 - \frac{1-\gamma}{2q+1}$ (the maximum is when the two terms equal), which is bounded away from 1.

If desired, the soundness error can be made arbitrarily close to γ by repeating the linearity test. Given a parameter t, the repeated linearity test samples $x_i, y_i \in \{0,1\}^\ell$ for each $i \in [t]$ and checks that $f(x_i) + f(y_i) = f(x_i + y_i)$ for all $i \in [t]$. Now, the PCP verifier makes $3t + 2q$ queries and has soundness error $\max\{(1-\varepsilon)^t, \gamma + 2q\varepsilon\}$, which for suitable ε and $t = O_{\gamma,\varepsilon}(q)$ is arbitrarily close to γ.

The non-signaling case. We follow the structure of the classical transformation. However, the non-signaling case not only calls for a different analysis but also raises a problem that we must solve.

The linearity test in the non-signaling setting has the following guarantee [CMS18]: if $\mathcal{F}$ is a k-non-signaling function such that $\Pr_{x,y,\mathcal{F}}[\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y)] \geq 1-\varepsilon$ then $\mathcal{F}$ can be self-corrected (in the natural way) to a $(k/2)$-non-signaling function $\hat{\mathcal{F}}$ that is $2^{O(k)}\varepsilon$-close to a *linear* non-signaling function $\mathcal{L}$. Note that self-correction is already part of the conclusion.

The above result appears sufficient for compiling a nsLPCP verifier into a corresponding nsPCP verifier. Namely, given a k-non-signaling function $\mathcal{F}\colon \{0,1\}^\ell \to \{0,1\}$, the nsPCP verifier checks that $\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y)$ for random $x, y \in \{0,1\}^\ell$ and also checks that the nsLPCP verifier accepts $\hat{\mathcal{F}}$. Analogously to before, if the nsLPCP verifier makes q queries and has soundness error γ against *linear* $(\frac{k-3}{2})$-non-signaling functions, then the resulting PCP verifier makes $3+2q$ queries and has soundness error $\max\{1-\varepsilon, \gamma + 2^{O(k)}\varepsilon\}$ against *arbitrary* k-non-signaling functions.

However, our analysis of the ALMSS verifier (the nsLPCP that we use) will require locality $k \geq \Omega(\log N)$, which means that the additive term $2^{O(k)}\varepsilon$ grows with N. This precludes achieving a constant soundness error with constant query complexity.

The foregoing motivates the problem of testing linearity of non-signaling functions *with low error*: how do we ensure that $\hat{\mathcal{F}}$ is sufficiently close to a linear non-signaling function $\mathcal{L}$? We stress that while in the classical case improving the "quality" of the self-correction has a straightforward solution (repeat the linearity test, and do self-correction), in the non-signaling case this problem is quite involved. Moreover, *we do not wish to modify in any way the classical compiler*, and thus relying on additional queries (even if only a constant number depending on q and ε) is not an option.

We discuss our solution to this problem later on in Section 2.6, thereby providing the missing ingredient of our compiler from nsLPCP to nsPCP. In the meantime, in Sections 2.2 to 2.5, we discuss how we prove that the ALMSS verifier is secure against linear non-signaling functions.

2.2 The linear ALMSS verifier against linear non-signaling functions

Our goal is to establish that the linear PCP verifier of [ALMSS98] (the "ALMSS verifier") is sound against linear non-signaling functions, and thus prove that every language $L \in \mathsf{DSIZE}(S)$ has a constant-query nsLPCP verifier that is sound against linear $O(\log S)$-non-signaling functions. Note that we invoke the ALMSS verifier on *deterministic* (DSIZE) computations, rather than on *nondeterministic* (NSIZE) computations as in the classical case. We now recall the ALMSS verifier.

Let $L \in \mathsf{DSIZE}(S)$ be a language, and let $\{C_n\}_{n\in\mathbb{N}}$ be a uniform boolean circuit family of size $N := S(n)$ that decides L (for all $x \in \{0,1\}^n$, $x \in L$ iff $C_n(x) = 1$). Hereafter we omit the subscript in C_n as it is clear from context. Given an input x, one can express the condition "$C(x) = 1$" as a system of simple equations over C's wires W; the variables are $\mathbf{w} = (w_1, \dots, w_N)$, one per wire. We use the convention that the input wires are $w_1, \dots, w_n$ and the output wire is w_N. To ensure input consistency we need that $w_j = x_j$ for every $j \in \{1, \dots, n\}$; to ensure correct gate computations we need that, for every $j \in \{n+1, \dots, N\}$, w_j is the correct combination of the variables used to compute it (e.g., if w_j is the output of an AND gate with inputs w_{j_1} and w_{j_2} then the equation is $w_j = w_{j_1} \cdot w_{j_2}$); to ensure that the output is 1 we need that $w_N = 1$. This can be summarized as a system of $M := N + 1$ equations $\{P_j(\mathbf{w}) = c_j\}_{j\in[M]}$, where $P_1, \dots, P_M$ are quadratic polynomials (each involving at most three variables in $\mathbf{w}$) and $c_1, \dots, c_M$ are boolean constants.

The ALMSS verifier is given below. We overload notation and use P_j to also denote the upper triangular matrix in $\{0,1\}^{N^2}$ such that $P_j(\mathbf{w}) = \langle P_j, \mathbf{w} \otimes \mathbf{w}\rangle$; that is, if $P_j(\mathbf{w}) = \sum_{i=1}^{N} a_i w_i + \sum_{1\le i<i'\le N} a_{i,i'} w_i w_{i'}$, then P_j has a_i in the diagonal entry (i,i) and $a_{i,i'}$ in the entry (i,i'), for $1 \le i < i' \le N$. Also, for $a \in \{0,1\}^N$, D_a is the diagonal matrix in $\{0,1\}^{N^2}$ whose diagonal is a.

The ALMSS verifier, given input $x \in \{0,1\}^n$ and oracle access to a linear non-signaling function $\mathcal{L}\colon \{0,1\}^{N^2} \to \{0,1\}$, works as follows:

1. Use the circuit C and input x to construct the matrices $P_1, \dots, P_M \in \{0,1\}^{N^2}$ and constants $c_1, \dots, c_M \in \{0,1\}$, which represent the computation of C on x.
2. Draw random $s \in \{0,1\}^M$, $u, v, \in \{0,1\}^N$, and query $\mathcal{L}$ on the set $\{\sum_{j=1}^{M} s_j P_j, D_u, D_v, u\otimes v\}$.
3. Check that $\mathcal{L}(\sum_{j=1}^{M} s_j P_j) = \sum_{j=1}^{M} s_j c_j$ and that $\mathcal{L}(D_u)\mathcal{L}(D_v) = \mathcal{L}(u \otimes v)$.

If $C(x) = 1$, the honest proof is the linear function $\pi\colon \{0,1\}^{N^2} \to \{0,1\}$ where $\pi(Z) := \langle \mathbf{w}\otimes\mathbf{w}, Z\rangle = \sum_{i,i'\in[N]} w_i w_{i'} \cdot Z_{i,i'}$ where w_i is now the value of the i-th wire in the computation of C on x.

The challenge is to prove that the ALMSS verifier is sound against *linear non-signaling functions*. Namely, we must show that if there is a linear non-signaling function $\mathcal{L}$ that is accepted with good probability then $x \in L$, or equivalently that $C(x) = 1$. We discuss this in the next sub-sections.

2.3 A linear local assignment generator suffices

The first step in our soundness analysis shows that, to establish that $C(x) = 1$, it suffices to construct a *linear local assignment generator* with sufficiently small error.

A linear k-local assignment generator for (C, x) with error ε is a linear k-non-signaling function $\mathcal{A}\colon \{0,1\}^N \to \{0,1\}$ that individually satisfies each of the M constraints with probability $1-\varepsilon$ (over the randomness of $\mathcal{A}$). Namely, (a) for each $i \in \{1, \dots, n\}$, $\Pr[\mathcal{A}(e_i) = x_i] \geq 1-\varepsilon$; (b) for each $i \in \{n+1, \dots, N\}$, if w_i is the output of a unary gate g with input w_j then $\Pr[\mathcal{A}(e_i) = g(\mathcal{A}(e_j))] \geq 1-\varepsilon$, else if w_i is the output of a binary gate g with inputs w_{j_1}, w_{j_2} then $\Pr[\mathcal{A}(e_i) = g(\mathcal{A}(e_{j_1}), \mathcal{A}(e_{j_2}))] \geq 1-\varepsilon$; (c) $\Pr[\mathcal{A}(e_N) = 1] \geq 1-\varepsilon$. (Here e_i is the i-th vector in the standard basis.)

Lemma (informal)**.** *If there exists a k-local assignment generator for (C, x) with error ε for $k = O(\log N)$ and $\varepsilon = O(\frac{1}{N \log N})$, then $C(x) = 1$.*

We sketch the proof of this lemma. The transcript of the computation of C on x is the *unique* correct assignment to all the wires. We say that a wire $w_i \in W$ of C is *correct* whenever $\mathcal{A}(e_i)$ equals the value contained in this transcript; more generally, we say that a vector $z \in \{0,1\}^N$ is correct if $\mathcal{A}(z)$ equals the value of z in the linear extension of the transcript. Below, we partition C's wires W into layers $W_1, \dots, W_H$ according to depth. (We assume layered circuits; see Section 3.1.)

As a warmup, suppose for now that $k \geq N$. The probability that all wires in W_1 are correct is at least $1 - |W_1|\,\varepsilon$. If we condition on all wires in $W_1 \cup \cdots \cup W_{h-1}$ being correct, then all wires in W_h are correct with probability $1 - |W_h|\,\varepsilon$. We deduce that the output wire is correct with probability $1 - \sum_{h=1}^{H} |W_h|\,\varepsilon = 1 - |W|\,\varepsilon = 1 - N\varepsilon$. Since the output wire is 1 with probability $1-\varepsilon$, and $\varepsilon = O(\frac{1}{N})$, we conclude that $\Pr[C(x) = 1] > 0$, and thus $C(x) = 1$.

The above argument requires that $k \geq N$, because we have to simultaneously "view" assignments to all wires in the circuit. While the argument can be easily modified so that we only require k to be at least twice the width of C, the latter may still be much larger than $O(\log N)$.

Using the linearity of $\mathcal{A}$, however, we can modify the argument to merely require $k = O(\log N)$. For each layer h, we define an event E_h such that if E_h holds, then any wire in layer h is correct with high probability. In the warmup above E_h is the event "all wires in layer $h-1$ are correct"; in our proof E_h is the event "t random linear combinations of wires in layer h are correct". Given a wire w_i in layer h, we can bound the event "$\mathcal{A}(e_i)$ is incorrect and E_h holds" as follows. If $\mathcal{A}(e_i)$ is incorrect, then all linear combinations of wires in layer h can be split into pairs z and $z + e_i$, and exactly one of $\mathcal{A}(z)$ and $\mathcal{A}(z + e_i)$ is incorrect. Hence, the probability that a random linear combination of wires in layer h is correct, given that $\mathcal{A}(e_i)$ is incorrect, is at most $1/2$, and so $\Pr[E_h \,|\, \mathcal{A}(e_i) \text{ is incorrect}] \leq 2^{-t}$, since the t random linear combinations are independent. Using Bayes's rule (and an additional assumption that $\Pr[E_h] \geq 1/2$), we deduce that $\Pr[\mathcal{A}(e_i) \text{ is incorrect} \,|\, E_h]$ is small. We then proceed inductively on the layers as before.

The argument above requires that $\varepsilon = O(\frac{1}{N \log N})$. One may wonder whether a similar result could be proved with, say, $\varepsilon = O(1)$. We additionally prove that our analysis is almost tight, in that an error of $\varepsilon = O(\frac{\log N}{N})$ is necessary, *regardless* of how large the locality k is.

See Section 7 for details.

Local assignment generators in prior works. Local assignment generators appear in prior works on nsPCPs [KRR14; PR17], but our notion is qualitatively different, as we now explain.

Prior works consider local assignment generators for an *augmented* circuit C_{aug} rather than for C itself. Informally, C_{aug} not only contains C as a sub-circuit but also low-degree extensions of C's layers as well as subcircuits computing all low-degree tests on these. The wires contained in these additional subcircuits are what enables defining an event E_h on which to condition for each layer.

The analogue of the augmented circuit C_{aug} in our setting, however, has *exponential* size, and thus *we cannot use it.* Namely, we would have to encode each layer of C via the Hadamard code (all linear combinations of wires in the layer) and then compute all possible linear tests on these.

Instead, our assumption that the local assignment generator is a *linear* non-signaling function implies that we *do not have to construct an augmented circuit.* Namely, the linear combinations that we use to define the event E_h are implicitly available due this linearity, and so there is no need to augment C (nor, in particular, to introduce any gates that evaluate linearity tests).

The assumption that the local assignment generator is linear is justified by the fact that a different part of our construction (the linearity test in our generic compiler) ensures the non-signaling function is (close to) linear. Overall, this separation not only avoids the aforementioned issues of using augmented circuits, but also simplifies the analysis of the local assignment generator.

2.4 Constructing the linear local assignment generator

Given a k-non-signaling function $\mathcal{L}\colon \{0,1\}^{N^2} \to \{0,1\}$ that is accepted by the ALMSS verifier with probability at least $1-\varepsilon$, we can obtain a linear k-local assignment generator $\mathcal{A}\colon \{0,1\}^N \to \{0,1\}$ with error $O(\varepsilon)$ by "restricting $\mathcal{L}$ to its diagonal". Namely, in order to query $\mathcal{A}$ at $v \in \{0,1\}^N$, we query $\mathcal{L}$ at $D_v \in \{0,1\}^{N^2}$, where D_v is the diagonal matrix that has v as its diagonal.

We show that, since $\mathcal{L}$ is accepted with probability at least $1-\varepsilon$, $\mathcal{L}$ must satisfy any *individual* constraint $P_j(\mathbf{w}) = c_j$ with probability at least $1 - O(\varepsilon)$, and this directly implies that the linear local assignment generator $\mathcal{A}$ has error $O(\varepsilon)$. (See Section 8.2 for details.)

The discussion so far already gives us a weak bound on the soundness error of the ALMSS verifier, namely $1 - O(\frac{1}{N \log N})$. Indeed, for $k = O(\log N)$ and $\varepsilon = O(\frac{1}{N \log N})$, we can apply the lemma above (in Section 2.3) to conclude that $C(x) = 1$.

However, our goal is to show that the ALMSS verifier (as is) has *constant* soundness error, and doing so requires more technical work, which we discuss next.

Remark 2.1. We stress that proving a soundness error of even $1 - O(\frac{1}{N \log N})$ is a non-trivial statement. This is in contrast to the classical setting, where if an assignment satisfies an $1-\varepsilon$ fraction of the $M = N + 1$ constraints for $\varepsilon < 1/M$, then, trivially, *all* constraints are satisfied.

2.5 The ALMSS verifier has soundness error $5/6$

Our goal is to prove that the ALMSS verifier has constant soundness error. In a first step (Section 2.5.1), we use the soundness error proved above (Section 2.4) to show that the t-repeated ALMSS verifier has soundness error γ when $t = O(\log N + \log \frac{1}{\gamma})$. In a second step (Section 2.5.2), we then prove that the basic ALMSS verifier (no repetitions) actually has constant soundness error.

2.5.1 The t-repeated ALMSS verifier has soundness error $\exp(-t)$

While in the classical setting reducing soundness error via simple repetition is straightforward (t-wise repetition reduces soundness error from δ to δ^t), in the non-signaling setting simple repetition *does not work.*[5] Indeed, consider the non-signaling function (in fact, distribution) that, with probability

[5] Even if simple repetition were to reduce soundness error from δ to δ^t, then to get $\delta^t = \gamma$ when $\delta = 1 - O(\frac{1}{N \log N})$ we would need to repeat $t = O(N \log N + \log \frac{1}{\gamma})$ times, which requires too large of a locality k for the analysis.

$1-\varepsilon$, answers the verifier's queries in an accepting way, and otherwise answers randomly. This non-signaling function is accepted by the t-repeated verifier with probability $\approx 1-\varepsilon$, which is about the same as the probability that it is accepted by a single verifier.

However, this example provides intuition for how one circumvents this issue. Informally, we would like to extract the "$1-\varepsilon$ good part" that satisfies the verifier, and drop the "ε bad part". We follow a technique used in [KRR14] and, instead of arguing about the probability that $\mathcal{L}$ passes the t-repeated verifier, we argue that the non-signaling function $\mathcal{L}$ *conditioned on passing the t-repeated verifier* passes the basic verifier with high probability. Indeed, in the aforementioned example, conditioning on at least one test passing removes the "ε bad part" injected by the distribution, and intuitively *extracts* the part of $\mathcal{L}$ that is passing the verifier. An interesting feature of our analysis of the verifier is that our conclusion is about the basic verifier, not the relaxed t-repeated verifier, which plays a major role in the analysis in [KRR14].[6] This is a qualitative difference in our analysis arising from our use of property testing (not present in [KRR14]), which also simplifies the analysis.

In more detail, let $\mathcal{L}'$ denote the linear non-signaling function that equals $\mathcal{L}$ when conditioned on passing the t-repeated verifier. Namely, if E is the (random) event that $\mathcal{L}$ passes the t-repeated verifier, then for any $S \subseteq \{0,1\}^n$ (of some maximal size) and $\vec{b} \in \{0,1\}^S$, we define

$$\Pr\left[\mathcal{L}'(S)=\vec{b}\right] := \Pr\left[\mathcal{L}(S)=\vec{b} \mid E\right] = \frac{\Pr[\mathcal{L}(S)=\vec{b} \wedge E]}{\Pr[E]} \quad . \tag{1}$$

We then prove that $\mathcal{L}'$ passes the basic verifier with probability at least $1-\frac{1/\Pr[E]}{\exp(t)}$.

The proof uses a generic lemma (Lemma 5.1) stating that, if we run $t+d$ independent tests, then the probability that at most r out of the first d tests pass and all of the last t tests pass is at most $(\frac{d}{t+d})^{r+1}$. A naive application of this lemma (with $r=0$ and $d=1$) shows that $\mathcal{L}'$ passes the basic verifier with probability at least $1-\frac{1/\Pr[E]}{(t+1)}$. This is not enough, because (since $\Pr[E] \geq \gamma$) we would require $t = O(N \log N \cdot \frac{1}{\gamma})$ to prove soundness, which is again far too many repetitions.

However, we leverage the linearity of $\mathcal{L}$ to deduce the stronger guarantee, as we now explain. We want to bound the probability that $\mathcal{L}'$ does not pass the basic verifier, which means we need to bound the probability that $\mathcal{L}$ fails exactly the first test of $t+1$ independent tests. We do this by arguing this individually for each of the two types of tests made by the ALMSS verifier: the tensor test "$\mathcal{L}(D_u)\mathcal{L}(D_v) = \mathcal{L}(u \otimes v)$" and the satisfiability test "$\mathcal{L}(\sum_{j=1}^{M} s_j P_j) = \sum_{j=1}^{M} s_j c_j$". We will explain our techniques in the case of the satisfiability test; the same techniques work for the tensor test, but the algebra is messier.

In the case of the satisfiability test, we split the "special" test (i.e., the first one) into d pairs of tests, such that each individual test is random, but each pair is correlated so that if both tests in some pair pass, then the original test passes. Specifically, we draw d random vectors $s^{(1)}, \ldots, s^{(d)} \in \{0,1\}^M$, and then we split the test "$\mathcal{L}(\sum_{j=1}^{M} s_j P_j) = \sum_{j=1}^{M} s_j c_j$" into the d pairs of tests

$$\text{“}\mathcal{L}\left(\sum_{j=1}^{M}(s_j + s_j^{(i)})P_j\right) = \sum_{j=1}^{M}(s_j + s_j^{(i)})c_j\text{”} \quad \text{and} \quad \text{“}\mathcal{L}\left(\sum_{j=1}^{M} s_j^{(i)} P_j\right) = \sum_{j=1}^{M} s_j^{(i)} c_j\text{”} \quad .$$

This allows us to apply the lemma with $d = O(t)$, and $r = O(t)$, which shows that $\mathcal{L}'$ passes the basic verifier with probability at least $1-\frac{1/\Pr[E]}{\exp(t)}$, an exponential decrease in t.

[6] The relaxed t-repeated verifier runs t tests and accepts if a large fraction of them pass.

The above analysis shows soundness error of $\frac{1}{\gamma}$ for the t-repeated verifier, for $t = O(\log N + \log \frac{1}{\gamma})$. Indeed, by the above argument, the conditioned function $\mathcal{L}'$ passes the basic verifier with probability $1 - \frac{1}{\gamma}\exp(-t) = 1 - O(\frac{1}{N \log N})$, by choice of t. The analysis in the previous section (Section 2.4) then implies that $C(x) = 1$, proving soundness of the t-repeated verifier.

The discussion so far merely shows that the t-wise repetition of the ALMSS verifier, which makes $4t$ queries, has a constant soundness error when $t = \Omega(\log N)$; moreover, we get no conclusions for $t = o(\log N)$. We still do not know what we can say about a *single* invocation of the 4-query ALMSS verifier. We next discuss how to handle this case.

2.5.2 Back to the 4-query ALMSS verifier

We establish that the ALMSS verifier has a constant soundness error by considering the t-repeated ALMSS verifier *only in the analysis*. In particular, our argument merely requires $k = O(t)$ but does not require at any point actually making $4t$ queries.

We start from the assumption that a linear k-non-signaling function $\mathcal{L}$ is accepted by the (4-query) ALMSS verifier with probability δ, for a sufficiently large constant $\delta \in (0, 1)$, and we wish to conclude that $C(x) = 1$. In the analysis (see below), we lower bound via a quantity $\gamma(\delta, t)$ the probability that $\mathcal{L}$ is accepted by the t-repeated ALMSS verifier and then follow Section 2.5.1 to construct another linear k'-non-signaling function $\mathcal{L}'$ that is accepted by the ALMSS verifier with probability $1 - 2^{-O(t)}$. Next, setting $t = O(\log N)$, we get that $\mathcal{L}'$ is accepted by the ALMSS verifier with probability $1 - O(\frac{1}{N \log N})$. Finally, we follow Section 2.4 by using $\mathcal{L}'$ to construct a linear local assignment generator with error $\varepsilon = O(\frac{1}{N \log N})$ and conclude that $C(x) = 1$.

We now describe the omitted step in the explanation above. We prove that if $\mathcal{L}$ is accepted by the ALMSS verifier with probability δ, then $\mathcal{L}$ is accepted by the t-repeated ALMSS verifier with probability at least $\gamma := \Omega((0.99\delta)^{t+1})$. (In fact, we prove a generic statement for any test; see Lemma 5.3.) While in the classical case (when $\mathcal{L}$ is merely a function), the probability of passing the t-repeated ALMSS verifier is *exactly* δ^t because the repetitions are independent, in the non-signaling case it is a priori not even clear whether a lower bound is possible because $\mathcal{L}$ can provide correlated answers across the repetitions. Nevertheless, we prove a lower bound that is *almost* δ^t (and, in particular, is almost tight).

Next, following Section 2.5.1, we construct $\mathcal{L}'$ (see Eq. (1)) that is accepted by the ALMSS verifier with probability at least $1 - \frac{1/\gamma}{\exp(t)}$. There is a slight complication, as while $\exp(t)$ grows exponentially in t, now $1/\gamma$ does also. We need $\exp(t)$ to grow faster than $1/\gamma$ does, which we achieve by carefully choosing δ ($\delta = 5/6$ suffices), and thereby get that $\mathcal{L}'$ is accepted by the ALMSS verifier with probability $1 - 2^{-O(t)}$.[7]

2.6 Testing linearity with low error

Below we discuss linearity testing *with low error* (Theorem 3) in more detail.

Warmup: distributions. We have discussed (in Section 2.1) how to test linearity with low error in the classical setting. In order to illustrate some of the difficulties that arise in the non-signaling setting, we first discuss a special case of it: testing linearity against a *distribution* over functions.

[7] One may believe that this step can be used to prove arbitrarily small soundness error for the ALMSS verifier (which is impossible). But the argument would fail, e.g., if we choose $\delta = o(1)$, as then $1/\gamma$ grows faster than $\exp(t)$.

First, suppose that $\mathcal{D}$ is a distribution over functions $f \colon \{0,1\}^n \to \{0,1\}$ that passes the linearity test with probability $1-\varepsilon$. The self-correction $\hat{\mathcal{D}}$ that on input $x \in \{0,1\}^n$ samples a random $z \in \{0,1\}^n$ and outputs $\hat{\mathcal{D}}(x) = \mathcal{D}(z+x) - \mathcal{D}(z)$ is 2ε-close to a distribution over *linear* functions $\mathcal{D}^*$, namely, for every $x \in \{0,1\}^n$ it holds that $\left|\Pr[\hat{\mathcal{D}}(x) = 1] - \Pr[\mathcal{D}^*(x) = 1]\right| \leq 2\varepsilon$. Indeed, consider the distribution $\mathcal{D}^*$ that samples $f \leftarrow \mathcal{D}$ and outputs any linear function f^* closest to f.[8] Then, for every function f and $x \in \{0,1\}^n$, the probability over a random $z \in \{0,1\}^n$ that $f^*(z) = f(z)$ and $f^*(z+x) = f(z+x)$ is at least $1 - 2\varepsilon_f$, where $\varepsilon_f := 1 - \Pr_{x,y}[f(x) + f(y) = f(x+y)]$. Denoting by d_f denotes the probability that $\mathcal{D}$ samples the function f, we conclude that $\left|\Pr[\mathcal{D}^*(x) = 1] - \Pr[\hat{\mathcal{D}}(x) = 1]\right| \leq \sum_f 2\varepsilon_f \cdot d_f = 2\varepsilon$.

Next, suppose that we seek a self-correction of $\mathcal{D}$ that is δ-close to a distribution over linear functions, for $\delta \ll 2\varepsilon$. One idea is to follow the same strategy as in the case of a single function: repeat the linearity test and then do self-correction. This idea, however, does not work now.

Consider the distribution $\mathcal{D} = (1-\varepsilon) \cdot \mathbf{0} + \varepsilon \cdot \mathbf{1}$, i.e., the distribution that with probability $1-\varepsilon$ answers according to the all-zeros function (a linear function), and with probability ε according to the all-ones function (a function maximally far from linear functions). While $\mathcal{D}$ passes the linearity test with probability $1-\varepsilon$, $\mathcal{D}$ also passes the t-repeated linearity test with probability $1-\varepsilon$. In other words, if $\mathcal{D}$ passes the t-repeated linearity test with probability $1-\varepsilon$, we can still only conclude that $\hat{\mathcal{D}}$ is 2ε-close to linear, independent of t.

While repeating the test does not increase the rejection probability, it can still be used to improve the quality of self-correction, by considering a *different* notion of self-correction that penalizes functions in the support of $\mathcal{D}$ that are far from linear. Concretely, consider the distribution $\mathcal{D}_t$ that equals $\mathcal{D}$ *when conditioned on the event that the t-repeated linearity test passes*, and then define $\hat{\mathcal{D}}_t$ to be the self-correction of $\mathcal{D}_t$. That is, $\hat{\mathcal{D}}_t$ samples f from $\mathcal{D}_t$ and answers any query $x \in \{0,1\}^n$ by sampling $z \in \{0,1\}^n$ and returning $f(z+x) - f(z)$. We claim that $\hat{\mathcal{D}}_t$ is very close to linear.

Indeed, suppose that $\mathcal{D}$ passes the t-repeated test with probability $\gamma > 0$, and let $c > 1$ be a parameter. A function f sampled from $\mathcal{D}_t$ is $\frac{\ln c}{t}$-close to linear with probability at least $\frac{\gamma - 1/c}{\gamma} = 1 - \frac{1}{\gamma c}$.[9] Setting $c := t/\log t$, the probability that $\mathcal{D}_t$ outputs a function f that is $\frac{\log t - \log\log t}{t}$-far from linear is at most $\frac{\log t}{\gamma t}$. Therefore, by applying the argument from the beginning of this subsection, we conclude that $\hat{\mathcal{D}}_t$ is $O_\gamma(\frac{\log t}{t})$-close to a distribution over linear functions.

We can further reduce the distance to be exponentially small in t by performing self-correction t times: $\hat{\mathcal{D}}_t(x)$ now samples $z_1, \dots, z_t \in \{0,1\}^n$, and outputs the majority of $\{\mathcal{D}(z_i + x) - \mathcal{D}(z_i)\}_{i \in [t]}$ conditioned on the event that the t-repeated linearity test passes. By setting $c := 2^{t/10}$ in the discussion above, we conclude that if we sample f from $\mathcal{D}_t$, then f is 0.1-close to a linear function f^* with probability $1 - \frac{1}{\gamma 2^{t/10}}$. In particular, for every $x \in \{0,1\}^n$ it holds that $\Pr_{z_i}[f^*(z_i) = f(z_i) \wedge f^*(z_i + x) = f(z_i + x)] \geq 0.8$, and so the probability that the majority value of $\{\mathcal{D}(z_i + x) - \mathcal{D}(z_i)\}_{i \in [t]}$ is not equal to $f^*(x)$ is $2^{-\Omega(t)}$. In sum, the t-repeated self-correction conditioned on the event that the t-repeated linearity test passes yields us a distribution that is $\frac{1}{\gamma} 2^{-\Omega(t)}$-close to linear.

The non-signaling case. The case of non-signaling strategies is similar to the case of distributions in that the analysis of the self-correction involves conditioning over a certain event. Yet, the conclusions and steps of the proof are quite different. Informally, this is because non-signaling functions are quasi-distributions (probabilities can be negative), which prevents us from doing a

[8] Recall that if f is 0.25-close to linear functions then f^* is unique. We do not rely on uniqueness.

[9] The t-repeated linearity test accepts a function f that is $\frac{\ln c}{t}$-far from linear with probability at most $(1 - \frac{\ln c}{t})^t \leq \frac{1}{c}$.

straightforward analysis such as the one above. We now discuss how we address this.

Suppose that we have a k-non-signaling function $\mathcal{F}\colon \{0,1\}^n \to \{0,1\}$ that passes the linearity test with probability $1-\varepsilon$. The result of [CMS18] proves that the self-correction $\hat{\mathcal{F}}$ defined as $\hat{\mathcal{F}}(x) := \mathcal{F}(z+x) - \mathcal{F}(z)$ (where z is chosen randomly from $\{0,1\}^n$) is $2^{O(k)}\varepsilon$-close to linear. This is too large in our setting as we have $k = O(\log N)$, and we would like the distance to be $O(\varepsilon)$. Instead, we prove a slightly different guarantee from [CMS18]. Namely, we show that there is a linear non-signaling function $\mathcal{L}$, such that on every set S, $\hat{\mathcal{F}}$ is $O(|S|\sqrt{\varepsilon})$-close to $\mathcal{L}$. Unlike in the result of [CMS18], our distance now decays with $|S|$, and is in particular independent of k. This is sufficient for our purposes, since we set $|S| = 4$, the number of queries made by the ALMSS verifier.

In our proof, we consider a *different* self-correction $\overline{\mathcal{F}}_t$ that, unlike $\hat{\mathcal{F}}$, is *only used in the analysis* and is not used by the compiler. First, we show that $\overline{\mathcal{F}}_t$ passes the linearity test with probability $1-\exp(-t)$, and so the result of [CMS18] implies that $\overline{\mathcal{F}}_t$ is very close to a linear non-signaling function. Then, we relate $\overline{\mathcal{F}}_t$ and $\hat{\mathcal{F}}$ to show that $\hat{\mathcal{F}}$ is $O(\sqrt{\varepsilon})$-close to a linear non-signaling function.

Informally, the self-correction $\overline{\mathcal{F}}_t$ equals $\mathcal{F}$ with the standard self-correction procedure repeated t times, conditioned on $\mathcal{F}$ passing $(1-\sqrt{\varepsilon})t$ of t repetitions of the linearity test. In more detail, given a subset $S \subseteq \{0,1\}^n$, $\overline{\mathcal{F}}_t(S)$ is the following distribution. For each $x \in S$, sample uniform and independent $z_x^{(1)}, \ldots, z_x^{(t)} \in \{0,1\}^n$ conditioned on satisfying the same linear dependencies as in S; for instance, if $S = \{x, y, x+y\}$, then $z_x^{(i)} + z_y^{(i)} = z_{x+y}^{(i)}$ holds for all i. Then $\overline{\mathcal{F}}_t$ assigns to each $x \in S$ the value $\mathrm{maj}_{i\in[t]}\{\mathcal{F}(z_x^{(i)}+x) - \mathcal{F}(z^{(i)})\}$ *conditioned on the event* that $\mathcal{F}$ passes at least $(1-\sqrt{\varepsilon})t$ of t repetitions of the basic linearity test. We note that if $\mathcal{F}$ is linear, then $\overline{\mathcal{F}}_t \equiv \hat{\mathcal{F}} \equiv \mathcal{F}$.

The first part of the analysis uses Lemma 5.2, which informally states that by conditioning on $\mathcal{F}$ passing most of the t-repeated linearity tests, we force the conditioned $\mathcal{F}$ to behave "close" to linear. Specifically, letting $b_x^{(i)} = \mathcal{F}(z_i + x) - \mathcal{F}(z_i)$, we get that with probability $1-\exp(-t)$ there is a bit b_x that equals $b_x^{(i)}$ for at least $\frac{3t}{4}$ of the i's (so the majority is a vast majority), which implies that $\overline{\mathcal{F}}_t(x) = b_x$, and analogously for y and $x+y$. Then, via a similar argument, we show that with probability $1-\exp(-t)$ for at least $\frac{3t}{4}$ of the i's it holds that $b_x^{(i)} + b_y^{(i)} = b_{x+y}^{(i)}$. By union bound, these events hold simultaneously, and so we conclude that $\overline{\mathcal{F}}_t$ satisfies $\overline{\mathcal{F}}_t(x) + \overline{\mathcal{F}}_t(y) = \overline{\mathcal{F}}_t(x+y)$ with probability $1-\exp(-t)$. We then invoke the result of [CMS18] and conclude that $\hat{\mathcal{F}}$ is very close to some linear non-signaling function $\mathcal{L}$.

In the second step, we relate $\hat{\mathcal{F}}$ to $\overline{\mathcal{F}}_t$ by claiming that if $\Pr[\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y)] \geq 1-\varepsilon$, then $\hat{\mathcal{F}}$ and $\overline{\mathcal{F}}_t$ are close in some precise sense (see Section 9.4 for details). We first observe that if we run the t-repeated linearity test, i.e., choose $x^{(1)}, y^{(1)}, \ldots, x^{(t)}, y^{(t)}$ and check that $\mathcal{F}(x^{(i)}) + \mathcal{F}(y^{(i)}) = \mathcal{F}(x^{(i)} + y^{(i)})$ for every i, then a simple Markov argument (Lemma 5.5) shows that with high probability, most of the linearity tests are satisfied. For instance, with probability $1-\sqrt{\varepsilon}$ at least $(1-\sqrt{\varepsilon})t$ of the i's satisfy the linear constraint. This means that the event conditioned on in the definition of $\overline{\mathcal{F}}_t$ is a large event. We also know from the first part of the analysis that, with high probability, the conditioning causes most of the evaluations of $\mathcal{F}(z_x^{(i)} + x) - \mathcal{F}(z_x^{(i)})$ to output the same value. Intuitively, this implies that $\hat{\mathcal{F}}$ is close to $\overline{\mathcal{F}}_t$, via the following reasoning. Since $\overline{\mathcal{F}}_t$ conditions on a large event, it is close to the corresponding self-correction that does not condition at all. Since the majority taken over the evaluations of $\mathcal{F}(z_x^{(i)} + x) - \mathcal{F}(z_x^{(i)})$ when computing $\overline{\mathcal{F}}_t$ is a vast majority, with high probability $\hat{\mathcal{F}}$ (which is a sample from one of the elements the majority is over) will agree with the vast majority. This allows us to conclude that for any set S, $\hat{\mathcal{F}}$ will be $O(|S|\sqrt{\varepsilon})$-close to $\overline{\mathcal{F}}_t$.

See Section 9 for details.

3 Definitions

We introduce the main definitions used throughout the paper.

3.1 Boolean circuits and the class $\mathsf{DSIZE}(S)$

All boolean circuits in this paper are layered and have maximum fan-in 2. In more detail, a boolean circuit is a layered directed acyclic graph, in which the vertices are called "gates" and the edges are called "wires". Furthermore, each gate is labeled as either AND, OR or NOT, and it holds that (1) every wire goes from a gate in layer h to a gate in layer $h+1$, (2) each gate has at most 2 incoming wires, (3) each gate with 2 incoming wires is either an AND gate or an OR gate, and (4) each gate with 1 incoming wire is a NOT gate.

The class $\mathsf{DSIZE}(S)$ is the main complexity class that we study in the paper.

Definition 3.1. *Given a computable function $S\colon \mathbb{N} \to \mathbb{N}$, a language L is in $\mathsf{DSIZE}(S)$ if there exists a uniform family of boolean circuits $C_n\colon \{0,1\}^n \to \{0,1\}$ such that C_n has at most $S(n)$ gates (and is constructible in time $\tilde{O}(S(n))$) and, for all $x \in \{0,1\}^n$, $x \in L$ if and only if $C_n(x) = 1$.*

The reader may find it helpful to compare $\mathsf{DSIZE}(S)$ with $\mathsf{NSIZE}(S)$, where the second condition is replaced by "$x \in L$ if and only if there exists w such that $C_n(x,w) = 1$".

3.2 Non-signaling functions

We define *non-signaling functions* and introduce useful notation for them. The text below is taken almost verbatim from [CMS18].

Definition 3.2. *A* **k-non-signaling function** *$\mathcal{F}\colon D \to \{0,1\}$ is a collection $\mathcal{F} = \{\mathcal{F}_S\}_{S \subseteq D, |S| \leq k}$ where (i) each $\mathcal{F}_S$ is a distribution over functions $f\colon S \to \{0,1\}$, and (ii) for every two subsets S and T each of size at most k, the restrictions of $\mathcal{F}_S$ and $\mathcal{F}_T$ to $S \cap T$ are equal as distributions. (If $S = \emptyset$ then $\mathcal{F}_S$ always outputs the empty string.)*

Note that any function $f\colon D \to \{0,1\}$ induces a $|D|$-non-signaling function by setting $\mathcal{F}_S$ to be the distribution that outputs $f|_S$ with probability 1. More generally, any distribution $\mathcal{D}$ over functions $f\colon D \to \{0,1\}$ induces a corresponding $|D|$-non-signaling function by defining $\mathcal{F}_S$ to be the distribution that samples $f \leftarrow \mathcal{D}$ and outputs $f|_S$.

Given a set $S \subseteq D$ of size $|S| \leq k$ and a string $\vec{b} \in \{0,1\}^S$, we define

$$\Pr\left[\mathcal{F}(S) = \vec{b}\right] := \Pr_{f \leftarrow \mathcal{F}_S}\left[f(S) = \vec{b}\right] \quad .$$

The non-signaling property in this notation is the following: for every two subsets $S, T \subseteq D$ of sizes $|S|, |T| \leq k$ and every string $\vec{b} \in \{0,1\}^{S \cap T}$, $\Pr[\mathcal{F}(S)|_{S \cap T} = \vec{b}] = \Pr[\mathcal{F}(T)|_{S \cap T} = \vec{b}]$, where the probability is over the randomness of $\mathcal{F}$.

Sometimes it is more convenient to consider a *vector* of inputs (rather than a *set* of inputs), and so we define notation for this case. Given a vector $(x_1, \ldots, x_s)$ with entries in D and a vector $(b_1, \ldots, b_s)$ with entries in $\{0,1\}$ (with $s \in \{1, \ldots, k\}$), we define $\Pr[\mathcal{F}((x_1, \ldots, x_s)) = (b_1, \ldots, b_s)]$ and $\Pr[\mathcal{F}(x_1) = b_1, \ldots, \mathcal{F}(x_s) = b_s]$ to be the probability

$$\Pr_{f \leftarrow \mathcal{F}_{\{x_1, \ldots, x_s\}}}\left[f(x_1) = b_1, \ldots, f(x_s) = b_s\right] \quad .$$

Note that $\{x_1, \ldots, x_s\}$ is an unordered set and its size may be less than s, because the entries of the vector $(x_1, \ldots, x_s)$ may not be distinct. We abuse notation and still use symbols such as S and $\vec{b}$ to denote vectors as above. We stress that we use an ordering on S merely to match each element of S to the corresponding element in $\vec{b}$; the event remains unchanged if one permutes the entries of S and $\vec{b}$ according to the same permutation.

3.3 Quasi-distributions

A quasi-distribution extends the notion of a probability distribution by allowing negative probabilities. Quasi-distributions are equivalent to non-signaling functions [CMS18], and are a useful to view, and analyze, non-signaling functions. The text below is taken almost verbatim from [CMS18].

Definition 3.3 (quasi-distributions). *Let D be a finite domain, and denote by U_D the set of all boolean functions of the form $f \colon D \to \{0,1\}$. A* **quasi-distribution** *$\mathcal{Q}$ over a subset $G \subseteq U_D$ is a set of real numbers $\{q_f\}_{f \in U_D}$ such that $\sum_{f \in U_D} q_f = 1$ and $q_f = 0$ for every $f \notin G$.*

Definition 3.4 (quasi-probability). *Given a quasi-distribution $\mathcal{Q} = \{q_f\}_{f \in U_D}$, a subset $S \subseteq D$, and a string $\vec{b} \in \{0,1\}^S$, we define the* **quasi-probability** *of the event "$\mathcal{Q}(S) = \vec{b}$" to be the following (possibly negative) real number*

$$\widetilde{\Pr}[\mathcal{Q}(S) = \vec{b}] := \sum_{f \in U_D \text{ s.t. } f(S) = \vec{b}} q_f \ .$$

As in the case of non-signaling functions, it is sometimes more convenient to consider a *vector* of inputs rather than a *set*. Given a vector $(x_1, \ldots, x_s)$ with entries in D and a vector $(b_1, \ldots, b_s)$ with entries in $\{0,1\}$, we define $\Pr[\mathcal{Q}((x_1, \ldots, x_s)) = (b_1, \ldots, b_s)]$ and $\Pr[\mathcal{Q}(x_1) = b_1, \ldots, \mathcal{Q}(x_s) = b_s]$ to be the (possibly negative) real number $\sum_{f \in U_D \text{ s.t. } \forall i \, f(x_i) = b_i} q_f$. We abuse notation and still use symbols such as S and $\vec{b}$ to denote vectors as above.

Definition 3.5 (locality). *Let D be a finite domain of size N and let $k \in \{1, \ldots, N\}$. A quasi-distribution $\mathcal{Q}$ over U_D is* **k-local** *if for every subset $S \subseteq D$ of size $|S| \le k$ and string $\vec{b} \in \{0,1\}^S$,*

$$\widetilde{\Pr}[\mathcal{Q}(S) = \vec{b}] \in [0,1] \ .$$

For completeness, we also say that all quasi-distributions are 0-local.

If $\mathcal{Q}$ is k-local, then for every subset $S \subseteq D$ of size $|S| \le k$, we may view $\mathcal{Q}(S)$ as a probability distribution over $\{0,1\}^S$. If $\mathcal{Q}$ is k-local then it is s-local for every $s \in \{0, 1, \ldots, k\}$.

For $\mathcal{Q}$ to be k-local, it suffices for all relevant $\widetilde{\Pr}[\mathcal{Q}(S) = \vec{b}]$ to be non-negative (as opposed to be in $[0,1]$). This is because $\sum_f q_f = 1$, so that $\sum_{\vec{b} \in \{0,1\}^S} \widetilde{\Pr}[\mathcal{Q}(S) = \vec{b}] = 1$ and, if all terms in this sum are non-negative, then we can deduce that $\widetilde{\Pr}[\mathcal{Q}(S) = \vec{b}] \le 1$ for every $\vec{b}$.

Definition 3.6 (statistical distance). *Given a finite domain D and an integer $k \in \{1, \ldots, |D|\}$, the* **Δ_k-distance** *between two quasi-distributions $\mathcal{Q}$ and $\mathcal{Q}'$ is*

$$\Delta_k(\mathcal{Q}, \mathcal{Q}') := \max_{S \subseteq D \,,\, |S| \le k} \Delta(\mathcal{Q}_S, \mathcal{Q}'_S) \ ,$$

where $\Delta(\mathcal{Q}_S, \mathcal{Q}'_S) := \max_{E \subseteq \{0,1\}^S} \left| \widetilde{\Pr}[\mathcal{Q}(S) \in E] - \widetilde{\Pr}[\mathcal{Q}'(S) \in E] \right|$.

We say that $\mathcal{Q}$ and $\mathcal{Q}'$ are **ε-close in the Δ_k-distance** *if $\Delta_k(\mathcal{Q}, \mathcal{Q}') \le \varepsilon$; else, they are ε-far.*

Remark 3.7 (distance for non-signaling functions)**.** The definition of Δ_k-distance naturally extends to defining distances between k'-non-signaling functions, as well as between quasi-distributions and k'-non-signaling functions, provided that $k \leq k'$.

The notion above generalizes the standard notion of statistical (total variation) distance: if $\mathcal{Q}$ and $\mathcal{Q}'$ are *distributions* then their $\Delta_{|D|}$-distance equals their statistical distance. Also note that if $\mathcal{Q}$ and $\mathcal{Q}'$ are k-local quasi-distributions then their Δ_k-distance equals the maximum statistical distance, across all subsets $S \subseteq D$ with $|S| \leq k$, between the two *distributions* $\mathcal{Q}_S$ and $\mathcal{Q}'_S$ — in particular this means that any experiment that queries exactly one set of size at most k cannot distinguish between the two quasi-distributions with probability greater than $\Delta_k(\mathcal{Q}, \mathcal{Q}')$.

We conclude by stating a theorem from [CMS18], which motivates the use of quasi-distributions. Informally, the theorem states that quasi-distributions and non-signaling functions are equivalent.

Theorem 4 ([CMS18])**.** *For any finite domain D, the following hold:*

1. *For every k-local quasi-distribution $\mathcal{Q}$ over functions $f \colon D \to \{0,1\}$ there exists a k-non-signaling function $\mathcal{F} \colon D \to \{0,1\}$ such that $\Delta_k(\mathcal{Q}, \mathcal{F}) = 0$.*

2. *For every k-non-signaling function $\mathcal{F} \colon D \to \{0,1\}$ there exists a k-local quasi-distribution $\mathcal{Q}$ over functions $f \colon D \to \{0,1\}$ such that $\Delta_k(\mathcal{Q}, \mathcal{F}) = 0$. The set of such quasi-distributions forms an affine subspace of $\mathbb{R}^{2^{|D|}}$.*

3.4 Input-oblivious queries

An oracle algorithm $A \colon \{0,1\}^* \to \{0,1\}$ has *input-oblivious queries* if, given an input $x \in \{0,1\}^n$, the queries to its oracle are determined non-adaptively based solely on n and on its internal randomness, and only the final decision depends on x. In more detail, one can view A as specified by a pair of algorithms (Q, D) where: (a) Q is a probabilistic algorithm known as the *query sampler* that, given n, outputs a set of queries for the oracle; (b) D is a deterministic algorithm known as the *decision predicate* that, given x (and Q's randomness), outputs a decision bit. All verifiers that we consider in this paper have input-oblivious queries.

3.5 Non-signaling PCPs

A *non-signaling PCP* (nsPCP) [KRR14] is a PCP in which soundness is further required to hold against any (sufficiently local) non-signaling function.

In this setting, for a query alphabet D and answer alphabet Σ, a nsPCP verifier is an algorithm V that, given oracle access to a non-signaling function $\mathcal{F} \colon D \to \Sigma$, queries $\mathcal{F}$ at a single subset $S \subseteq D$ (on which $\mathcal{F}$ is defined), receives answers $a \in \Sigma^S$, and outputs a decision bit.

Definition 3.8. *Given $\varepsilon \in [0,1]$ and $k \in \mathbb{N}$, a nsPCP verifier V for a language L has soundness error ε against k-non-signaling functions if the following holds for every $x \in \{0,1\}^*$:*
1. If $x \in L$ then there exists a function $\pi \colon D \to \Sigma$ such that $\Pr[\mathsf{V}^{\pi}(x) = 1] = 1$.
2. If $x \notin L$ then, for every k-non-signaling function $\mathcal{F} \colon D \to \Sigma$, $\Pr[\mathsf{V}^{\mathcal{F}}(x) = 1] \leq \varepsilon$.

Remark 3.9. In Item 2, if we consider only all functions $\pi \colon D \to \Sigma$, rather than all k-non-signaling functions $\mathcal{F} \colon D \to \Sigma$, then we recover the usual PCP definition.

If the query alphabet D has size at most ℓ, the PCP verifier uses at most r random bits to sample a subset S of size at most q in time T_1, and the PCP verifier decides to accept/reject in time T_2 based on $\mathcal{F}$'s answers to S, then we write

$$L \in \mathsf{nsPCP}\begin{bmatrix} \textsf{soundness error:} & \varepsilon \\ \textsf{randomness:} & r \\ \textsf{proof length:} & \ell \\ \textsf{query complexity:} & q \\ \textsf{locality:} & k \\ \textsf{query sampler time:} & T_1 \\ \textsf{decision predicate time:} & T_2 \end{bmatrix} .$$

Note that we must require $q \leq k$, since querying $\mathcal{F}$ on a set of size greater than k is undefined.

3.6 Linear non-signaling PCPs

A *linear non-signaling PCP* (nsLPCP) is a PCP that is sound against (sufficiently local) non-signaling functions that are *linear*, that is, they are induced by quasi-distributions over linear functions. This definition is motivated by the equivalence between non-signaling functions and quasi-distributions, as well as the linearity testing result, proved in [CMS18].

Definition 3.10. *A k-non-signaling function $\mathcal{L}\colon \{0,1\}^\ell \to \{0,1\}$ is* linear *if there exists a k-local quasi-distribution $\mathcal{Q}$ over linear functions $f\colon \{0,1\}^\ell \to \{0,1\}$ such that $\Delta_k(\mathcal{L},\mathcal{Q}) = 0$.*

Definition 3.11. *Given $\varepsilon \in [0,1]$ and $k \in \mathbb{N}$, a nsLPCP verifier V for a language L has soundness error ε against linear k-non-signaling functions if the following holds for every $x \in \{0,1\}^*$:*
1. If $x \in L$ then there exists a linear function $\pi\colon \{0,1\}^\ell \to \{0,1\}$ such that $\Pr[\mathsf{V}^\pi(x) = 1] = 1$.
2. If $x \notin L$ then, for every linear k-non-signaling function $\mathcal{L}\colon \{0,1\}^\ell \to \{0,1\}$, $\Pr[\mathsf{V}^{\mathcal{L}}(x) = 1] \leq \varepsilon$.

Remark 3.12. In Item 2, if we only consider all linear functions $\pi\colon \{0,1\}^\ell \to \{0,1\}$, rather than all linear k-non-signaling functions $\mathcal{L}\colon \{0,1\}^\ell \to \{0,1\}$, then we recover the usual LPCP definition.

If the LPCP verifier uses at most r random bits to sample a subset S of size at most q in time T_1, queries $\mathcal{L}\colon \{0,1\}^\ell \to \{0,1\}$ on S, and decides to accept/reject in time T_2 based on $\mathcal{L}$'s answers to S, then we write

$$L \in \mathsf{nsLPCP}\begin{bmatrix} \textsf{soundness error:} & \varepsilon \\ \textsf{randomness:} & r \\ \textsf{proof length:} & \ell \\ \textsf{query complexity:} & q \\ \textsf{locality:} & k \\ \textsf{query sampler time:} & T_1 \\ \textsf{decision predicate time:} & T_2 \end{bmatrix} .$$

Like before, we must require $q \leq k$, since querying $\mathcal{L}$ on a set of size greater than k is undefined.

4 Formal statements of our results

Our main theorem (discussed in Section 1.2) is that the exponential-length constant-query PCP in [ALMSS98] (with no modifications) is sound against non-signaling functions. We obtain:

Theorem 5 (formal statement of Theorem 1)**.**

$$\mathsf{DSIZE}(S) \subseteq \mathsf{nsPCP}\left[\begin{array}{rl} \textit{soundness error:} & 1 - 1/10^5 \\ \textit{randomness:} & O(S^2) \\ \textit{proof length:} & 2^{O(S^2)} \\ \textit{query complexity:} & 11 \\ \textit{locality:} & O(\log^2 S) \\ \textit{query sampler time:} & O(S^2) \\ \textit{decision predicate time:} & O(n) \end{array}\right] .$$

Our proof relies on several results (discussed in Section 1.3), which we formally state below. At the end of this section we show how to combine these results to prove Theorem 5.

First, we prove that the linear PCP in [ALMSS98] is sound against *linear* non-signaling functions.

Theorem 6 (formal statement of Theorem 2; proved in Section 8)**.** *The linear PCP in [ALMSS98] is sound against linear non-signaling functions, giving us the following class inclusion:*

$$\mathsf{DSIZE}(S) \subseteq \mathsf{nsLPCP}\left[\begin{array}{rl} \textit{soundness error:} & 5/6 \\ \textit{randomness:} & O(S) \\ \textit{proof length:} & O(S^2) \\ \textit{query complexity:} & 4 \\ \textit{locality:} & O(\log S) \\ \textit{query sampler time:} & O(S^2) \\ \textit{decision predicate time:} & O(n) \end{array}\right] .$$

Next, we provide a linearity test for non-signaling functions *with low error.* The statement involves the notion of self-correction for a non-signaling function, which we define first.

Definition 4.1. *The* **self-correction** *of a k-non-signaling function $\mathcal{F}\colon \{0,1\}^n \to \{0,1\}$ is the $\lfloor k/2 \rfloor$-non-signaling function $\hat{\mathcal{F}}\colon \{0,1\}^n \to \{0,1\}$ defined as follows. Given a set $\{x_1, \ldots, x_s\} \subseteq \{0,1\}^n$ with $s \le \lfloor k/2 \rfloor$, $\hat{\mathcal{F}}_{\{x_1,\ldots,x_s\}}$ samples uniform and independent $z_1, \ldots, z_s \in \{0,1\}^n$ and answers each x_i with $\mathcal{F}(z_i + x_i) - \mathcal{F}(z_i)$.*

Theorem 7 (formal statement of Theorem 3; proved in Section 9)**.** *Let $k, \bar{k} \in \mathbb{N}$ and $\varepsilon \in (0, 1/400]$ be such that $k \ge \Omega(\frac{\bar{k}}{\varepsilon} \cdot (\bar{k} + \log \frac{1}{\varepsilon}))$. Suppose that $\mathcal{F}\colon \{0,1\}^n \to \{0,1\}$ is a k-non-signaling function such that $\Pr_{x,y,\mathcal{F}}[\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y)] \ge 1 - \varepsilon$. Then there exists a linear $\bar{k}$-non-signaling function $\mathcal{L}\colon \{0,1\}^n \to \{0,1\}$ such that for all query sets $Q \subseteq \{0,1\}^n$ with size $|Q| \le \bar{k}$ and for all events $E \subseteq \{0,1\}^Q$ it holds that*

$$\left|\Pr[\hat{\mathcal{F}}(Q) \in E] - \Pr[\mathcal{L}(Q) \in E]\right| \le (6\,|Q| + 3)\sqrt{\varepsilon} \ .$$

The above result enables us to compile any nsLPCP into a corresponding nsPCP, via the usual transformation that runs a linearity test and then uses self-correction to ask the nsLPCP queries.

Lemma 4.2 (formal statement of Lemma 1.3; proved in Section 6)**.** *The classical LPCP-to-PCP compiler works in the non-signaling setting, giving us that, for every $\varepsilon \in (0,1)$,*

$$\mathsf{nsLPCP}\left[\begin{array}{rl} \textit{soundness error:} & 1-\varepsilon \\ \textit{randomness:} & r \\ \textit{proof length:} & \ell \\ \textit{query complexity:} & q \\ \textit{locality:} & k \\ \textit{query sampler time:} & T_1 \\ \textit{decision predicate time:} & T_2 \end{array}\right] \subseteq \mathsf{nsPCP}\left[\begin{array}{rl} \textit{soundness error:} & 1-\min\{\frac{1}{400},\frac{\varepsilon^2}{(6q+4)^2}\} \\ \textit{randomness:} & r+(q+2)\ell \\ \textit{proof length:} & 2^{\ell} \\ \textit{query complexity:} & 2q+5 \\ \textit{locality:} & O(\frac{k}{\varepsilon}(k+\log\frac{1}{\varepsilon})) \\ \textit{query sampler time:} & T_1+O(q\ell) \\ \textit{decision predicate time:} & T_2+O(q) \end{array}\right],$$

Proof of Theorem 5. By invoking the transformation from Lemma 4.2 on the ALMSS verifier from Theorem 6, we directly obtain the inclusion stated in Theorem 5. □

5 Preliminaries on repeated tests against non-signaling functions

We state and prove several generic facts about the acceptance probability of repeated tests against non-signaling functions.

A q-query test over a domain D is a pair (R, Π), where R is a distribution over q-size subsets of D, and $\Pi \colon \{0,1\}^q \to \{0,1\}$ is a predicate; the test first samples $(x_1, \dots, x_q)$ according to R, makes the queries to $\mathcal{F}$, receives answers $(a_1, \dots, a_q) \in \{0,1\}^q$, and outputs $\Pi(a_1, \dots, a_q)$.

Given a k-non-signaling function $\mathcal{F} \colon D \to \{0,1\}$ and a q-query test over D, for any $t \leq \lfloor k/q \rfloor$ we can perform the test t times in parallel and thereby define t boolean random variables $X_1, \dots, X_t$ by setting $X_i = 1$ if the i-th test passes. These boolean variables are identically distributed but need not be independent, and their joint distribution is *permutation invariant* (for every $\vec{b} \in \{0,1\}^t$ and permutation $\pi \colon [t] \to [t]$ it holds that $\Pr[X_i = b_i \; \forall\, i \in [t]] = \Pr[X_{\pi(i)} = b_i \; \forall\, i \in [t]]$).

The lemmas below hold for any identically distributed boolean variables whose joint distribution is permutation-invariant.

Lemma 5.1 (adapted from [KRR14, Claim 6.2])**.** *Let $r, d, t \in \mathbb{N}$ be such that $0 \leq r < d$. Let $X_1, \dots, X_{t+d}$ be boolean random variables with a permutation-invariant joint distribution. Define the following events:*

- *H^d_{d-r} is the event that $X_i = 1$ for at least $d - r$ indices $i \in \{1, \dots, d\}$.*
- *T^t_t is the event that $X_i = 1$ for all $i \in \{d+1, \dots, d+t\}$.*

Then

$$\Pr_{X_1,\dots,X_{t+d}}\left[\neg H^d_{d-r} \wedge T^t_t\right] \leq \frac{\binom{t+d-r-1}{t}}{\binom{t+d}{t}} \leq \left(\frac{d}{t+d}\right)^{r+1} \quad .$$

In particular, if $\Pr\left[T^t_t\right] > 0$ *then*

$$\Pr_{X_1,\dots,X_{t+d}}\left[\neg H^d_{d-r} \mid T^t_t\right] \leq \frac{1}{\Pr\left[T^t_t\right]} \cdot \left(\frac{d}{t+d}\right)^{r+1} \quad .$$

Proof. Letting $I := \{i \in [t+d] : X_i = 0\}$, we view the event $\neg H^d_{d-r} \wedge T^t_t$ as happening in two steps. First, we sample $X_1, \dots, X_{t+d}$, which determines $|I|$. Second, we re-index the samples by selecting a uniformly random permutation $\pi \colon [t+d] \to [t+d]$ to determine I. This implies that

$$\Pr_{X_1,\dots,X_{t+d}}\left[\neg H^d_{d-r} \wedge T^t_t\right] = \sum_{j=0}^{t+d} \Pr_{\pi}\left[\neg H^d_{d-r} \wedge T^t_t \;\middle|\; |I| = j\right] \cdot \Pr_{X_1,\dots,X_{t+d}}[|I| = j] \quad .$$

The above step uses the fact that the joint distribution of $X_1, \dots, X_{t+d}$ is permutation-invariant. If $j \leq r$, then $\Pr_\pi[\neg H^d_{d-r} \wedge T^t_t \mid |I| = j] = 0$, because in this case H^d_{d-r} always holds. If $j \geq r+1$, then $\Pr_\pi[\neg H^d_{d-r} \wedge T^t_t \mid |I| = j] = \binom{t+d-j}{t} / \binom{t+d}{t}$, because the event holds if and only if $\pi(d+1), \dots, \pi(d+t)$ are all not in I, which happens with probability $\binom{t+d-j}{t} / \binom{t+d}{t}$ when $|I| = j$. Hence, $\Pr_\pi[\neg H^d_{d-r} \wedge T^t_t \mid |I| = j] \leq \binom{t+d-r+1}{t} / \binom{t+d}{t}$, for $j \geq r+1$. Therefore,

$$\begin{aligned}
&\Pr_{X_1,\dots,X_{t+d}}[\neg H^d_{d-r} \wedge T^t_t] \\
&\leq \sum_{j=r+1}^{t+d} \frac{\binom{t+d-(r+1)}{t}}{\binom{t+d}{t}} \cdot \Pr_{X_1,\dots,X_{t+d}}[|I| = j]
\end{aligned}$$

$$\leq \frac{\binom{t+d-r-1}{t}}{\binom{t+d}{t}} \quad . \qquad \square$$

Next we prove a variant of Lemma 5.1, where we consider the event that $X_i = 1$ for *most* indices i in $\{d+1, \dots, d+t\}$, as opposed to *all* indices i in that set.

Lemma 5.2. *Let $d, t \in \mathbb{N}$, and let $0 \leq \varepsilon < \delta \leq 1$. Let $X_1, \dots, X_{t+d}$ be boolean random variables with a permutation-invariant joint distribution. Define the following events:*

- *$H^d_{(1-\delta)d}$ is the event that $X_i = 1$ for at least $(1-\delta)d$ indices $i \in \{1, \dots, d\}$.*
- *$T^t_{(1-\varepsilon)t}$ is the event that $X_i = 1$ for at least $(1-\varepsilon)t$ indices $i \in \{d+1, \dots, d+t\}$.*

Then,

$$\Pr_{X_1,\dots,X_{t+d}}\left[\neg H^d_{(1-\delta)d} \wedge T^t_{(1-\varepsilon)t}\right] \leq e^{-\frac{1}{8}(\delta-\varepsilon)^2 \min\{t,d\}} \quad .$$

In particular, if $\Pr\left[T^t_{(1-\varepsilon)t}\right] > 0$ *then*

$$\Pr_{X_1,\dots,X_{t+d}}\left[\neg H^d_{(1-\delta)d} \mid T^t_{(1-\varepsilon)t}\right] \leq \frac{1}{\Pr\left[T^t_{(1-\varepsilon)t}\right]} \cdot e^{-\frac{1}{8}(\delta-\varepsilon)^2 \min\{t,d\}} \quad .$$

Proof. The proof strategy follows that for Lemma 5.1. Letting $I := \{i \in [t+d] : X_i = 0\}$, we view the event $\neg H^d_{(1-\delta)d} \wedge T^t_{(1-\varepsilon)t}$ as happening in two steps. First, we sample $X_1, \dots, X_{t+d}$, which determines $|I|$. Second, we re-index the samples by selecting a uniformly random permutation $\pi \colon [t+d] \to [t+d]$ to determine I. This implies that

$$\Pr_{X_1,\dots,X_{t+d}}\left[\neg H^d_{(1-\delta)d} \wedge T^t_{(1-\varepsilon)t}\right] = \sum_{j=0}^{t+d} \Pr_{\pi}[\neg H^d_{(1-\delta)d} \wedge T^t_{(1-\varepsilon)t} \mid |I| = j] \cdot \Pr_{X_1,\dots,X_{t+d}}[|I| = j] \quad .$$

The above step uses the fact that the joint distribution of $X_1, \dots, X_{t+d}$ is permutation-invariant. By standard martingale concentration bounds (see, e.g., [McD98]), for $j < \frac{\varepsilon+\delta}{2}(t+d)$ it holds that

$$\Pr\left[\neg H^d_{(1-\delta)d} \mid |I| = j\right] \leq e^{-\frac{1}{2}(\delta - \frac{j}{t+d})^2 d} \leq e^{-\frac{1}{8}(\delta-\varepsilon)^2 d} \quad ,$$

and for $j \geq \frac{\varepsilon+\delta}{2}(t+d)$ it holds that

$$\Pr\left[T^t_{(1-\varepsilon)t} \mid |I| = j\right] \leq e^{-\frac{1}{2}(\frac{j}{t+d} - \varepsilon)^2 t} \leq e^{-\frac{1}{8}(\delta-\varepsilon)^2 t} \quad .$$

Therefore, all probabilities $\Pr_\pi\left[\neg H^d_{(1-\delta)d} \wedge T^t_{(1-\varepsilon)t} \mid |I| = j\right]$ are upper bounded by the maximum among the two bounds, which completes the proof of the lemma. □

We now state and prove two lemmas that lower bound the probability that $X_1 = \cdots = X_t = 1$ in terms of the probability that $X_i = 1$. Both lemmas play a key role in our results and, to the best of our knowledge, have not appeared in prior work studying non-signaling functions.

Suppose that $\Pr[X_i = 1] = \delta$. If the X_i's are independent, then $\Pr[X_1 = \cdots = X_t = 1] = \delta^t$. The same implication is not true when the X_i's are arbitrarily correlated. Yet, if their joint distribution is permutation-invariant, then we can conclude that δ^t (approximately) lower bounds this probability.

Lemma 5.3. *Let $X_1, \ldots, X_r$ be boolean random variables with a permutation-invariant joint distribution. Let $t \in \mathbb{N}$ and $0 < \tau < \delta < 1$ be such that $r \geq \frac{4t}{\tau}$ and $\Pr[X_i = 1] = \delta$ for every $i \in [r]$. Then, for some absolute constant $c > 0$,*

$$\Pr_{X_1,\ldots,X_t}[X_1 = \cdots = X_t = 1] \geq c \cdot \tau \cdot (\delta - \tau)^t \ .$$

Proof. Let $X = \sum_{i=1}^r X_i$. We have that $\mathbb{E}[X] = \sum_{i=1}^r \mathbb{E}[X_i] = r\delta$. Since the joint distribution of the X_i's is permutation-invariant, the probability that $X_1, \ldots, X_t$ are all 1 can be viewed as the probability that $X_i = 1$ for all i in a random subset of t X_i's out of the r X_i's. Therefore,

$$\Pr_{X_1,\ldots,X_t}[X_1 = \cdots = X_t = 1] = \sum_{i=0}^{r} \Pr[X = i] \cdot \frac{\binom{i}{t}}{\binom{r}{t}} = \mathbb{E}\left[\frac{\binom{X}{t}}{\binom{r}{t}}\right] \ .$$

Below we lower bound the foregoing expression.

Since $\mathbb{E}[X] = r\delta$ we have $\Pr[X \geq (\delta - \tau/2)r] \geq \tau/2$. Therefore,

$$\Pr_{X_1,\ldots,X_t}[X_1 = \cdots = X_t = 1] = \mathbb{E}\left[\frac{\binom{X}{t}}{\binom{r}{t}}\right] \geq \tau/2 \cdot \frac{\binom{(\delta-\tau/2)r}{t}}{\binom{r}{t}} \ .$$

It remains to lower bound the above binomial coefficient. We do this using the following claim.

Claim 5.4. *Let $\beta \in [0,1]$ and $r, t \in \mathbb{N}$ such that $r/t \geq 2/\beta$. Then $\frac{\binom{\beta r}{t}}{\binom{r}{t}} \geq c_0 \cdot (\beta - 2t/r)^t$ for some absolute constant $c_0 \in [0,1]$.*

Proof. By Stirling's approximation we have

$$c_0 \leq \frac{\binom{n}{i}}{\sqrt{\frac{n}{i\cdot(n-i)}} \cdot \frac{n^n}{i^i \cdot (n-i)^{n-i}}} \leq 1$$

for some absolute constant $c_0 > 0$. Therefore

$$\frac{\binom{\beta r}{t}}{\binom{r}{t}} \geq c_0 \cdot \frac{\sqrt{\frac{\beta r}{t\cdot(\beta r - t)}} \cdot \frac{(\beta r)^{\beta r}}{t^t (\beta r - t)^{\beta r - t}}}{\sqrt{\frac{r}{t \cdot (r-t)}} \cdot \frac{r^r}{t^t (r-t)^{r-t}}} = c_0 \cdot \frac{(1 - \frac{t}{r})^r}{(1 - \frac{t}{\beta r})^{\beta r}} \cdot \frac{(\beta r - t)^t}{(r-t)^t}$$

Using the assumption that $0 < \frac{t}{r} < \frac{t}{\beta r} < 1$ and the fact that $\frac{1-x}{e} \leq (1-x)^{1/x} \leq \frac{1-x/2}{e}$ for all $x \in (0,1]$, we get that

$$\frac{\binom{\beta r}{t}}{\binom{r}{t}} \geq c_0 \cdot \left(\frac{\frac{1}{e} - \frac{t}{er}}{\frac{1}{e} - \frac{t}{2e\beta r}}\right)^t \cdot \left(\beta - \frac{t}{r}\right)^t \geq c_0 \cdot \left(\frac{1 - \frac{t}{r}}{1 - \frac{t}{2\beta r}}\right)^t \cdot \left(\beta - \frac{t}{r}\right)^t \geq c_0 \cdot \left(\beta - 2\frac{t}{r}\right)^t$$

where the last inequality uses the assumption that $r/t \geq 2/\beta$. □

Using Claim 5.4, we get that

$$\Pr_{X_1,\ldots,X_t}[X_1 = \cdots = X_t = 1] \geq \tau/2 \cdot c_0 \cdot (\delta - \tau/2 - 2t/r)^t \geq c \cdot \tau \cdot (\delta - \tau)^t \ ,$$

where the last inequality uses the fact that $r \geq \frac{4t}{\tau}$. □

Finally, we prove a variant of Lemma 5.3, where we consider the event that most of the first t X_i's are 1, as opposed to all X_i's.

Lemma 5.5. *Let $X_1, \dots, X_t$ be boolean random variables with a permutation-invariant joint distribution, and $\rho, \delta \in [0,1]$ be such that $\Pr[X_i = 1] \geq 1 - \delta\rho$. Then*

$$\Pr_{X_1,\dots,X_t}[X_i = 1 \textit{ for at least } (1-\rho)t \textit{ indices } i \in [t]] \geq 1 - \delta \ .$$

Proof. Let $X = t - \sum_{i=1}^{t} X_i$. Observe that $\mathbb{E}[X] \leq \rho\delta t$, and so $\Pr[X \geq \rho t] \leq \mathbb{E}[X]/\rho t = \delta$. This implies that $\Pr[X < \rho t] \geq 1 - \delta$, and the lemma follows. □

6 From linear PCPs to standard PCPs

In this section we use Theorem 7 (testing linearity with low error) to prove Lemma 4.2, which shows that any non-signaling *linear* PCP can be compiled into a corresponding non-signaling PCP. This is the non-signaling analogue of the classical compiler from linear PCPs to PCPs [ALMSS98].

6.1 The PCP verifier

Let $\mathsf{V}_{\mathsf{LPCP}} = (\mathsf{Q}_{\mathsf{LPCP}}, \mathsf{D}_{\mathsf{LPCP}})$ be a linear PCP verifier that has oracle access to a linear k-non-signaling function $\mathcal{L}\colon \{0,1\}^\ell \to \{0,1\}$. We need to construct a PCP verifier $\mathsf{V}_{\mathsf{PCP}} = (\mathsf{Q}_{\mathsf{PCP}}, \mathsf{D}_{\mathsf{PCP}})$ that has oracle access to a k_{PCP}-non-signaling function $\mathcal{F}\colon \{0,1\}^\ell \to \{0,1\}$.

The construction of $\mathsf{V}_{\mathsf{PCP}}$ from $\mathsf{V}_{\mathsf{LPCP}}$ is *exactly the same* as in the classical case. That is, $\mathsf{V}_{\mathsf{PCP}}$ runs the [BLR93] linearity test, and then runs $\mathsf{V}_{\mathsf{LPCP}}$, simulating its oracle by querying the proof in a self-correcting manner. More specifically, for every query $a \in \{0,1\}^\ell$ that $\mathsf{V}_{\mathsf{LPCP}}$ makes to its oracle, $\mathsf{V}_{\mathsf{PCP}}$ samples a random $z_a \in \{0,1\}^\ell$, queries $\mathcal{F}(z_a), \mathcal{F}(z_a + a)$, and answers $\mathsf{V}_{\mathsf{LPCP}}$ with $\mathcal{F}(z_a + a) - \pi(z_a)$. Overall we can express $\mathsf{V}_{\mathsf{PCP}} = (\mathsf{Q}_{\mathsf{PCP}}, \mathsf{D}_{\mathsf{PCP}})$ in the following way:

> The query sampler $\mathsf{Q}_{\mathsf{PCP}}$ works as follows:
> 1. Sample uniformly random $x, y \in \{0,1\}^\ell$.
> 2. Run $\mathsf{Q}_{\mathsf{LPCP}}$ to obtain a set of queries $S_{\mathsf{LPCP}} \subseteq \{0,1\}^\ell$ for the linear PCP.
> 3. For each $a \in S_{\mathsf{LPCP}}$ sample $z_a \in \{0,1\}^\ell$ independently and uniformly at random.
> 4. Output the query set $S_{\mathsf{PCP}} = \{x, y, x+y\} \cup \{z_a, z_a + a : a \in S_{\mathsf{LPCP}}\}$
>
> Given the answers $\mathcal{F}(S_{\mathsf{PCP}})$ from the proof, define $\hat{\mathcal{F}}(S_{\mathsf{LPCP}})$ by letting $\hat{\mathcal{F}}(a) = \mathcal{F}(z_a + a) - \mathcal{F}(a)$ for all $a \in S_{\mathsf{LPCP}}$. The decision predicate $\mathsf{D}_{\mathsf{PCP}}$ accepts if and only if $\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y)$ and $\mathsf{D}_{\mathsf{LPCP}}(\hat{\mathcal{F}}(S_{\mathsf{LPCP}})) = 1$.

6.2 Parameters

The parameters of the new verifier $\mathsf{V}_{\mathsf{PCP}}$ follow from those of $\mathsf{V}_{\mathsf{LPCP}}$ and Theorem 7.

Randomness: $\mathsf{V}_{\mathsf{PCP}}$ uses 2ℓ random bits for the linearity test, plus additional $r + q\ell$ random bits to simulate $\mathsf{V}_{\mathsf{LPCP}}$. The total number of random bits is $r + (q+2)\ell$.

Length: The length is 2^ℓ, because the honest proof in the completeness case is a (classical) function $\pi\colon \{0,1\}^\ell \to \{0,1\}$, which takes 2^ℓ bits to write down.

Queries: $\mathsf{V}_{\mathsf{PCP}}$ makes 2 queries for every query made by $\mathsf{V}_{\mathsf{LPCP}}$, plus an additional 3 queries for the linearity test. The total number of queries is 11.

Locality: By Theorem 7 we need locality $k_{\mathsf{PCP}} \geq O(\frac{k_{\mathsf{LPCP}}}{\varepsilon} \cdot (k_{\mathsf{LPCP}} + \log \frac{1}{\varepsilon}))$.

Input obliviousness: The linearity test clearly does not depend on the input (or even L), and the self-correction only depends on its input set S_{LPCP}. Hence, if $\mathsf{V}_{\mathsf{LPCP}}$ has input-oblivious queries, then so does $\mathsf{V}_{\mathsf{PCP}}$.

Query sampler time: The query sampler of $\mathsf{V}_{\mathsf{PCP}}$ samples uniformly random $x, y \in \{0,1\}^\ell$. Then, it runs the query sampler of $\mathsf{V}_{\mathsf{LPCP}}$ to generate q queries, and finally generates the self-corrected queries. This takes a total of $T_1 + O(q\ell)$ time.

Decision predicate time: The decision predicate of $\mathsf{V}_{\mathsf{PCP}}$ first runs the linearity test predicate. Then, it computes the self-corrected answers to the queries of $\mathsf{V}_{\mathsf{LPCP}}$, and finally runs the decision predicate of $\mathsf{V}_{\mathsf{LPCP}}$. This takes a total of $T_2 + O(q)$ time.

6.3 Completeness

Completeness is straightforward. Suppose that there exists a linear function $\mathcal{F}\colon \{0,1\}^\ell \to \{0,1\}$ such that $\mathsf{V}_{\mathsf{LPCP}}^{\mathcal{F}}$ always accepts. We claim that $\mathcal{F}$ is also accepted by $\mathsf{V}_{\mathsf{PCP}}$ with probability 1. Indeed, if $\mathcal{F}$ is linear then $\hat{\mathcal{F}} \equiv \mathcal{F}$, and hence $\Pr[\mathsf{D}_{\mathsf{LPCP}}(\hat{\mathcal{F}}) = 1] = \Pr[\mathsf{D}_{\mathsf{LPCP}}(\mathcal{F}) = 1]$. Therefore,

$$\begin{aligned}\Pr[\mathsf{V}_{\mathsf{PCP}}^{\mathcal{F}} = 1] &= \Pr[\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y) \wedge \mathsf{D}_{\mathsf{LPCP}}(\hat{\mathcal{F}}(\mathsf{Q}_{\mathsf{LPCP}}))] \\ &= \Pr[\mathsf{D}_{\mathsf{LPCP}}(\mathcal{F}(\mathsf{Q}_{\mathsf{LPCP}})) = 1] \\ &= 1 \ .\end{aligned}$$

6.4 Soundness

Suppose that there exists a k_{PCP}-non-signaling function $\mathcal{F}\colon \{0,1\}^\ell \to \{0,1\}$ that is accepted by $\mathsf{V}_{\mathsf{PCP}}$ with probability at least $1-\gamma$ for some $\gamma \leq \min\{\frac{1}{400}, \frac{\varepsilon^2}{(6q+4)^2}\}$. In particular, this implies that $\Pr[\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y)] \geq 1 - \gamma \geq 1 - \frac{1}{400}$. Let $\mathcal{L}$ be the linear $\bar{k}$-non-signaling function from Theorem 7. Then,

$$\Pr[\mathsf{V}_{\mathsf{LPCP}}^{\mathcal{L}} = 1] \geq \Pr[\mathsf{V}_{\mathsf{LPCP}}^{\hat{\mathcal{F}}} = 1] - (6q+3)\sqrt{\gamma} \geq 1 - \gamma - (6q+3)\sqrt{\gamma} \geq 1 - \varepsilon \ ,$$

since $\gamma \leq \frac{\varepsilon^2}{(6q+4)^2}$, and so the soundness of $\mathsf{V}_{\mathsf{LPCP}}$ implies soundness of $\mathsf{V}_{\mathsf{PCP}}$.

7 Linear local assignment generators

In this section we define the notion of a *linear* local assignment generator, and prove that, in order to establish soundness of an LPCP, it suffices to construct a linear local assignment generator with sufficiently small error. This is one of the steps in the proof of Theorem 6 in Section 8, and we state it separately because it works for any nsLPCP (not just the ALMSS verifier analyzed in Section 8).

Informally, a linear local assignment generator for a circuit C on input x is a linear non-signaling function that satisfies each local constraint in the computation of C on x with high probability. This definition augments earlier ones in [KRR14; PR17] with the linearity condition (see Section 2.3).

Definition 7.1. *Let $C\colon \{0,1\}^n \to \{0,1\}$ be a boolean circuit with N wires, and let $x \in \{0,1\}^n$ be an input for it. A* **linear k-local assignment generator for (C, x) with error** *ε is a linear k-non-signaling function $\mathcal{A}\colon \{0,1\}^N \to \{0,1\}$ that satisfies the following.*

1. The input wires $w_1, \ldots, w_n$ are consistent with x: *for every $i \in [n]$,* $\Pr[\mathcal{A}(e_i) = x_i] \geq 1 - \varepsilon$.

2. The output wire w_N is 1: $\Pr[\mathcal{A}(e_N) = 1] \geq 1 - \varepsilon$.

3. AND/OR gates compute correctly: *for every binary gate g with inputs w_{i_1}, w_{i_2} and output w_j,*

$$\Pr[g(\mathcal{A}(e_{i_1}), \mathcal{A}(e_{i_2})) = \mathcal{A}(e_j)] \geq 1 - \varepsilon \ .$$

4. NOT gates compute correctly: *for every unary gate g with input w_i and output w_j,*

$$\Pr[g(\mathcal{A}(e_i)) = \mathcal{A}(e_j)] \geq 1 - \varepsilon \ .$$

We prove that the existence of a linear k-local assignment generator for (C, x), with sufficiently large locality k and sufficiently small error ε, implies that $C(x) = 1$. This tells us that, in order to prove soundness for a nsLPCP, it suffices to show that the existence of a linear non-signaling function accepted by the nsLPCP verifier with good probability implies the existence of a linear local assignment generator with low error.

Lemma 7.2. *Let $C\colon \{0,1\}^n \to \{0,1\}$ be a boolean circuit with N wires, and let $x \in \{0,1\}^n$ be an input for it. If there exists a linear k-local assignment generator $\mathcal{A}$ for (C, x) with error ε, where $k \geq 2\log_2(1/\varepsilon)$ and $\varepsilon \log_2(1/\varepsilon) \leq \frac{1}{20N}$, then $C(x) = 1$.*

The bound on the error is almost tight, *regardless* of the locality k.

Lemma 7.3. *For sufficiently large N and any n, there exist a boolean circuit $C\colon \{0,1\}^n \to \{0,1\}$ with N wires and an input $x \in \{0,1\}^n$ such that $C(x) = 0$ and yet exists a linear 2^N-local assignment generator $\mathcal{A}\colon \{0,1\}^N \to \{0,1\}$ for (C, x) with error $\varepsilon \geq \frac{\ln N}{N}$.*

We first prove Lemma 7.2 in Section 7.1, and then prove Lemma 7.3 in Section 7.2.

7.1 Proof of Lemma 7.2

Fix a boolean circuit C (see Section 3.1) with W wires, and an input $x \in \{0,1\}^n$ for C; denote by H the number of layers in C. For $h \in [H]$, let W_h be the set of wires in layer h, and let $D_h = \mathrm{span}\{e_i : w_i \in W_h\}$. In other words, D_h is the subset of $\{0,1\}^N$ containing all vectors v such that $v_i = 0$ for all indices $i \in [N]$ where the wire w_i is not in layer h. There is a unique correct transcript of the wires of the circuit for the computation of C on x; we let $\mathrm{Tr}_{(C,x)}\colon \{0,1\}^N \to \{0,1\}$ denote the linear extension of this transcript.

For a set $S \subseteq \{0,1\}^N$ let $E(S)$ be the event that $\mathcal{A}(S)$ returns values that are consistent with the correct transcript. That is, $E(S)$ occurs if and only if for each $\alpha \in S$ it holds that $\mathcal{A}(\alpha) = \langle \mathrm{Tr}_{(C,x)}, \alpha \rangle$. When the set $S = \{\alpha_1, \ldots, \alpha_t\}$ is specified explicitly by its elements, we allow ourselves to write $E(\alpha_1, \ldots, \alpha_t)$ to refer to $E(\{\alpha_1, \ldots, \alpha_t\})$.

Let $t = \lfloor \log_2(1/\varepsilon) \rfloor$ be a parameter. By our choice of parameters, $k \geq 2t \geq t+2$. For each layer $h \in [H]$ of the circuit, let $z_{h,1}, \ldots, z_{h,t} \in D_h$ be chosen uniformly at random. That is, each $z_{h,i}$ is a uniformly random linear combination of the wires in layer h of C. For a layer $h \in [H]$ of the circuit C, let E_h denote the event that $E(z_{h,1}, \ldots, z_{h,t})$ holds, i.e., all values output by $\mathcal{A}(z_{h,1}, \ldots, z_{h,t})$ are consistent with the correct transcript of C on input x. Note that the randomness of E_h is over both $z_{h,1}, \ldots, z_{h,t}$ and over $\mathcal{A}$; E_h will be the event that we will condition on in layer h (see Section 2.3).

In the proof we use the following definition.

Definition 7.4. *A set $S \subseteq \{0,1\}^N$ (of linear combinations of wires) is p-***good for layer** *h if $\Pr[E(S) \mid E_h] \geq p$. When $S = \{\alpha\}$ is a singleton, we say that α is p-good for layer h.*

Note that in the foregoing definition we do not insist that the elements of S belong to D_h, and it is well-defined to ask whether a linear combination of wires outside D_h is good for layer h.

We first argue that if E_h occurs with probability at least 0.5, then *any* linear combination of wires in layer h is p-good for p close to 1.

Claim 7.5. *If $\Pr[E_h] \geq 0.5$, then every $\alpha_h \in D_h$ is $(1 - 2^{-t+1})$-good for layer h.*

Proof. Fix $\alpha_h \in D_h$ and let $\beta_1, \ldots, \beta_t \in D_h$ be chosen independently and uniformly at random. For $i = 1, \ldots, t$, pick $z_{h,i} \in \{\beta_i, \alpha_h + \beta_i\}$ uniformly; note that the $z_{h,i}$'s are uniform in D_h.

We claim that $\Pr[E(z_{h,1}, \ldots, z_{h,t}) \mid \neg E(\alpha_h)] \leq 2^{-t}$. We first observe that if $E(\alpha_h)$ does not hold then $\mathcal{A}(\alpha_h)$ is not consistent with the transcript $\mathrm{Tr}_{(C,x)}$, i.e., $\mathcal{A}(\alpha_h) \neq \langle \mathrm{Tr}_{(C,x)}, \alpha_h \rangle$. Therefore, if $k \geq t+1$, then by the linearity of $\mathcal{A}$ for each $i \in [t]$ it holds that either $\mathcal{A}(\beta_i) \neq \langle \mathrm{Tr}_{(C,x)}, \beta_i \rangle$ or $\mathcal{A}(\alpha_h + \beta_i) \neq \langle \mathrm{Tr}_{(C,x)}, \alpha_h + \beta_i \rangle$. In other words, if $E(\alpha_h)$ does not hold, then either $E(\beta_i)$ does not hold or $E(\alpha_h + \beta_i)$ does not hold for all $i \in [t]$ independently of each other.[10] Hence, $\Pr[E(z_{h,1}, \ldots, z_{h,t}) \mid \neg E(\alpha_h)] \leq 2^{-t}$.

Therefore, since $k \geq t+1$, we have

$$\begin{aligned}
\Pr[\neg E(\alpha_h) \mid E_h] &= \frac{\Pr[\neg E(\alpha_h) \wedge E_h]}{\Pr[E_h]} \\
&\leq 2 \Pr[\neg E(\alpha_h) \wedge E(z_{h,1}, \ldots, z_{h,t})] \quad (\text{since } \Pr[E_h] \geq 0.5) \\
&\leq 2 \Pr[E(z_{h,1}, \ldots, z_{h,t}) \mid \neg E(\alpha_h)] \\
&\leq 2^{-t+1} \ .
\end{aligned}$$

By definition, this means that α_h is $(1 - 2^{-t+1})$-good for layer h, as claimed. □

We now use the above claim, and the condition that $\mathcal{A}$ satisfies the constraints representing the computation of C on x with high probability, to show that if E_h occurs with probability at least 0.5, then any linear combination of wires in the *next* layer is good with high probability.

Claim 7.6. *If $\Pr[E_h] \geq 0.5$, then every $\alpha_{h+1} \in D_{h+1}$ is $(1 - (2^{-t+2} + 2\varepsilon) \cdot N_{h+1})$-good for layer h.*

[10] This is the only place in the proof where we use the fact that the $z_{h,i}$'s are random.

Proof. Fix $\alpha_{h+1} \in D_{h+1}$. If $\alpha_{h+1} = e_i$ is a standard basis vector with 1 in a coordinate corresponding to some wire w_i in W_{h+1}, then there is some gate g in C taking two inputs w_{i_1}, w_{i_2} in layer h and outputting w_i in layer $h+1$. By Claim 7.5 (and union bound), we have

$$\Pr[E(\{e_{i_1}, e_{i_2}\}) \mid E_h] \geq 1 - 2^{-t+2} \ .$$

By the assumption that $\mathcal{A}$ is a local assignment generator for (C, x) with error ε it holds that

$$\Pr[g(\mathcal{A}(e_{i_1}), \mathcal{A}(e_{i_2})) = \mathcal{A}(e_i)] \geq 1 - \varepsilon \ ,$$

which implies that

$$\Pr[\neg E(e_i) \wedge E(\{e_{i_1}, e_{i_2}\})] \leq \Pr[\neg E(e_i) \mid E(\{e_{i_1}, e_{i_2}\})] \leq \varepsilon \ .$$

Hence,

$$\begin{aligned}
\Pr[\neg E(e_i) \mid E_h] = &\ \Pr[\neg E(e_i) \wedge \neg E(\{e_{i_1}, e_{i_2}\}) \mid E_h] + \Pr[\neg E(e_i) \wedge E(\{e_{i_1}, e_{i_2}\}) \mid E_h] \\
\leq &\ \Pr[\neg E(\{e_{i_1}, e_{i_2}\}) \mid E_h] + \Pr[\neg E(e_i) \wedge E(\{e_{i_1}, e_{w_2}\}) \mid E_h] \\
\leq &\ 2^{-t+2} + \frac{\Pr[\neg E(e_i) \wedge E(\{e_{i_1}, e_{i_2}\})]}{\Pr[E_h]} \\
\leq &\ 2^{-t+2} + 2\varepsilon \quad (\text{since } \Pr[E_h] \geq 0.5) \ .
\end{aligned}$$

Therefore, for $\alpha_{h+1} = e_i$ it holds that $\mathcal{A}(\alpha_{h+1})$ conditioned on E_h is consistent with $\mathrm{Tr}_{(C,x)}$ with probability at least $1 - (2^{-t+2} + 2\varepsilon)$, i.e., such α_{h+1} is $\big(1 - (2^{-t+2} + 2\varepsilon)\big)$-good for layer h.

If α_{h+1} is not a standard basis vector, then it is a linear combination of at most N_{h+1} basis vectors, and since $\mathcal{A}$ is a linear non-signaling function we get by union bound that $\mathcal{A}(\alpha_{h+1})$ is consistent with $\mathrm{Tr}_{(C,x)}$ with probability at least $1 - (2^{-t+2} + 2\varepsilon) \cdot N_{h+1}$, as required. □

We now use the foregoing claims to prove Lemma 7.2.

We use induction on the depth of C to show that E_h holds with high probability for every layer h. Specifically, we show that, for every layer h, $\Pr[E_h] \geq 1 - (2^{-t+2} + 2\varepsilon) \cdot \left(\sum_{i=1}^{h} N_i\right) \cdot t$. In particular, this will imply that $\Pr[E_H] \geq 0.5$, which will allow us to conclude that $C(x) = 1$.

Note that by the choice of the parameters, the obtained bound on $\Pr[E_h]$ will be greater than 0.5 for every layer $h \in [H]$, which will allow us to apply Claim 7.6 in the induction step. Indeed, for $k \geq 2\log_2(1/\varepsilon)$, $\varepsilon \log_2(1/\varepsilon) \leq \frac{1}{20N}$, and $t = \lfloor \log_2(1/\varepsilon) \rfloor \geq \log_2(1/\varepsilon) - 1$, we have

$$1 - (2^{-t+2} + 2\varepsilon) \cdot \left(\sum_{i=1}^{h} N_i\right) \cdot t \geq 1 - (2^{-t+2} + 2\varepsilon) \cdot N \cdot t \geq 1 - 10\varepsilon \cdot N \log_2(1/\varepsilon) \geq 0.5 \ .$$

We now provide the details of the induction and corresponding conclusion.

Base case. For the base case, we show that E_1 holds with probability $1 - (2^{-t+2} + 2\varepsilon) \cdot N_1 \cdot t$. Fix $i \in [n]$ so that w_i is the i-th input wire in C. Then, by the assumption that $\mathcal{A}$ is a linear local assignment generator for (C, x) with error ε we have $\Pr[\mathcal{A}(e_i) = x_i] \geq 1 - \varepsilon$. By the linearity of $\mathcal{A}$ and union bound, if $\alpha \in D_1$ then

$$\Pr[\mathcal{A}(\alpha) = \langle \mathrm{Tr}_{(C,x)}, \alpha \rangle] \geq 1 - \varepsilon n = 1 - \varepsilon N_1 \geq 1 - (2^{-t+2} + 2\varepsilon) \cdot N_1 \cdot t \ ,$$

which proves the base case of the induction.

Induction step. For the induction step we assume that $\Pr[E_h] \geq 1 - (2^{-t+2} + 2\varepsilon)(\sum_{i=1}^{h} N_i)t$, and prove the corresponding statement for E_{h+1}. By the earlier discussion we have $\Pr[E_h] \geq 0.5$, and thus by applying Claim 7.6 we get that for all $z_{h+1} \in D_{h+1}$ it holds that $\Pr[E(z_{h+1}) \mid E_h] \geq 1 - (2^{-t+2} + 2\varepsilon) \cdot N_{h+1}$. Therefore, for any $z_{h+1,1}, \ldots, z_{h+1,t} \in D_{h+1}$, by union bound we have

$$\Pr[E_{h+1} \mid E_h] = \Pr[E(\{z_{h+1,1}, \ldots, z_{h+1,t}\}) \mid E_h] \geq 1 - (2^{-t+2} + 2\varepsilon) \cdot N_{h+1} \cdot t \ .$$

This implies that

$$\begin{aligned}\Pr[E_{h+1}] \geq\ & \Pr[E_{h+1} \mid E_h] \cdot \Pr[E_h] \\ &= (1 - (2^{-t+2} + 2\varepsilon) \cdot N_{h+1} \cdot t) \cdot \Pr[E_h] \\ &\geq \left(1 - (2^{-t+2} + 2\varepsilon) \cdot N_{h+1} \cdot t\right) \left(1 - (2^{-t+2} + 2\varepsilon) \cdot \sum_{i=1}^{h} N_i \cdot t\right) \\ &\geq 1 - (2^{-t+2} + 2\varepsilon) \cdot \sum_{i=1}^{h+1} N_i \cdot t \ ,\end{aligned}$$

which proves the induction step.

Proving that $C(x) = 1$. It remains to show that $\Pr[E_H] \geq 0.5$ implies that $C(x) = 1$. Let w_N be the output wire of C, and let e_N be the corresponding basis vector. By Claim 7.5 we have $\Pr[\mathcal{A}(e_N) = C(x) \mid E_H] \geq 1 - 2^{-t+1}$. This implies that

$$\begin{aligned}\Pr[\mathcal{A}(e_N) = C(x)] \geq\ & \Pr[\mathcal{A}(e_N) = C(x) \wedge E_H] \\ =\ & \Pr[\mathcal{A}(e_N) = C(x) \mid E_H] \cdot \Pr[E_H] \\ \geq\ & (1 - 2^{-t+1})/2 \\ \geq\ & 1/4 \ .\end{aligned}$$

On the other hand, by the assumption that $\mathcal{A}$ is a linear local assignment generator for (C, x) with error ε, we have $\Pr[\mathcal{A}(e_N) \neq 1] \leq \varepsilon$. Therefore,

$$\begin{aligned}\Pr[C(x) \neq 1] =\ & \Pr[C(x) \neq \mathcal{A}(e_N) \wedge \mathcal{A}(e_N) = 1] + \Pr[C(x) = \mathcal{A}(e_N) \wedge \mathcal{A}(e_N) \neq 1] \\ \leq\ & \Pr[C(x) \neq \mathcal{A}(e_N)] + \Pr[\mathcal{A}(e_N) \neq 1] \\ \leq\ & 3/4 + \varepsilon \\ <\ & 1 \ .\end{aligned}$$

Since $C(x)$ is a deterministic value, we conclude that $C(x) = 1$, thus proving Lemma 7.2.

7.2 Proof of Lemma 7.3

Let $C \colon \{0,1\}^n \to \{0,1\}$ be a boolean circuit with only OR gates and let the input x be 0^n; note that $C(x) = 0$. Fix $\varepsilon > 0$. We first define a distribution $\mathcal{D}$ over functions $f \colon [N] \to \{0,1\}$.

To sample a function f from $\mathcal{D}$, we assign values to f one layer at a time. For each wire w_i in layer 0 (the input layer), $f(i) = x_i$ with probability $1 - \varepsilon$. Then, for each w_i in layer $h+1$, letting w_{i_1} and w_{i_2} be the wires in layer h that are inputs to the gate g that computes w_i, $f(i)$ is sampled as follows. If either $f(i_1)$ or $f(i_2)$ is 1, then $f(i) = 1$ with probability 1. Otherwise, $f(i)$ is 0 with probability $1 - \varepsilon$, and 1 with probability ε. In other words, if at least one of the input wires is 1, then the gate is computed correctly, otherwise it is computed correctly with probability $1 - \varepsilon$.

Given the distribution $\mathcal{D}$, we define $\mathcal{A} \colon \{0,1\}^N \to \{0,1\}$ to be the linear extension of $\mathcal{D}$ to the domain $\{0,1\}^N$. Namely, while $\mathcal{D}$ is a distribution over functions $f \colon [N] \to \{0,1\}$, $\mathcal{A}$ is the corresponding distribution over linear functions $h_f \colon \{0,1\}^N \to \{0,1\}$ such that h_f is the linear function defined by $h_f(e_i) = f(i)$ for all $i \in [N]$.

Note that $\mathcal{A}$ is a 2^N-non-signaling function (since it is a distribution), and is linear since it is a distribution over linear functions. Moreover, $\mathcal{A}$ satisfies all gate constraints locally with probability at least $1 - \varepsilon$, since $\mathcal{D}$ either computes the gate correctly with probability $1 - \varepsilon$, or it computes the gate correctly (but with incorrect inputs) with probability 1. Also, $\mathcal{A}$ assigns correct values to the input wires with probability $1 - \varepsilon$, since $\mathcal{D}$ does. Hence, to show that $\mathcal{A}$ is a linear local assignment generator with error ε, it remains to show that $\Pr[\mathcal{A}(e_N) = 1] \geq 1 - \varepsilon$.

We have that $\Pr[\mathcal{A}(e_N) = 1] = 1 - (1 - \varepsilon)^N$, since $\mathcal{A}$ assigns 1 to the output gate if at least one wire has the wrong value, and the probability that all wires are correct is $(1 - \varepsilon)^N$.

For $\varepsilon \geq \frac{\ln N}{N}$ and N sufficiently large, $1 - (1 - \varepsilon)^N \geq 1 - \varepsilon$. This is because $\frac{(1-\varepsilon)^N}{\varepsilon} \leq e^{-\varepsilon N - \ln \varepsilon} \leq e^{-\ln N - \ln \ln N + \ln N} = e^{-\ln \ln N} < 1$ for N sufficiently large.

In sum, $\mathcal{A}$ is a 2^N-local assignment generator for C on input x with error ε, but $C(x) = 0$.

8 The linear PCP verifier of ALMSS

In this section we prove Theorem 6.

Throughout, we fix a language $L \in \mathsf{DSIZE}(S)$ and an instance $x \in \{0,1\}^n$. We denote by $C := C_n$ the boolean circuit with $N := S(n)$ wires such that $x \in L$ if and only if $C(x) = 1$.

8.1 The ALMSS verifier

We analyze the LPCP verifier of [ALMSS98] (the "ALMSS verifier"), which we now recall.

The computation of C on x is viewed as a system of $M := N + 1$ equations $\{P_j(\mathbf{w}) = c_j\}_{j \in [M]}$ over N boolean variables $\mathbf{w} = (w_1, \ldots, w_N) \in \{0,1\}^N$, where $P_1, \ldots, P_M \colon \{0,1\}^N \to \{0,1\}$ are quadratic polynomials (each involving at most three variables in $\mathbf{w}$) and $c_1, \ldots, c_M$ are boolean constants. Each variable represents the value of one of the wires of C on input x; $w_1, \ldots, w_n$ are the n input wires and w_N is the output wire. There are three types of constraints:

Input consistency: For every $j \in \{1, \ldots, n\}$, $P_j(\mathbf{w}) := w_i$ and $c_j := x_j$.

Gate consistency: For every $j \in \{n+1, \ldots, N\}$,

- If the wire represented by the variable w_j is an output of an AND gate g where the inputs to g are given by w_{j_1}, w_{j_2}, then $P_j(\mathbf{w}) := w_j - w_{j_1} \cdot w_{j_2}$ and $c_j := 0$.
- If the wire represented by the variable w_j is an output of an OR gate g where the inputs to g are given by w_{j_1}, w_{j_2}, then $P_j(\mathbf{w}) := w_j - w_{j_1} - w_{j_2} + w_{j_1} \cdot w_{j_2}$ and $c_j := 0$.
- If the wire represented by the variable w_j is an output of a NOT gate g where the input to g is given by w_{j_1}, then $P_j(\mathbf{w}) := w_j - w_{j_1}$ and $c_j := 1$.

Accepting output: $P_M(\mathbf{w}) := w_N$ and $c_M := 1$.

We overload notation and use P_j to also denote the upper triangular matrix in $\{0,1\}^{N^2}$ such that $P_j(\mathbf{w}) = \langle P_j, \mathbf{w} \otimes \mathbf{w} \rangle$; that is, if $P_j(\mathbf{w}) = \sum_{i=1}^{N} a_i w_i + \sum_{1 \le i < i' \le N} a_{i,i'} w_i w_{i'}$, then P_j has a_i in the diagonal entry (i,i) and $a_{i,i'}$ in the entry (i,i'), for $1 \le i < i' \le N$. Also, for $a \in \{0,1\}^N$, D_a is the diagonal matrix in $\{0,1\}^{N^2}$ whose diagonal is a. The LPCP verifier of [ALMSS98] is as follows.

The ALMSS verifier, given input $x \in \{0,1\}^n$ and oracle access to a linear non-signaling function $\mathcal{L} \colon \{0,1\}^{N^2} \to \{0,1\}$, works as follows:

1. Use the circuit C and input x to construct the matrices $P_1, \ldots, P_M \in \{0,1\}^{N^2}$ and constants $c_1, \ldots, c_M \in \{0,1\}$, which represent the computation of C on x.
2. Sample $u, v \in \{0,1\}^N$ and $s \in \{0,1\}^M$ uniformly and independently at random.
3. Query the oracle $\mathcal{L}$ on the 4-element set $S = \{D_u, D_v, u \otimes v, \sum_{j=1}^{M} s_j P_j\}$.
4. Check that $\mathcal{L}(D_u)\mathcal{L}(D_v) = \mathcal{L}(u \otimes v)$ and $\mathcal{L}(\sum_{j=1}^{M} s_j P_j) = \sum_{j=1}^{M} s_j c_j$.

We also analyze the t-wise repetition of the ALMSS verifier (the "t-repeated ALMSS verifier").

Parameters. The parameters of the ALMSS verifier are the usual ones, which we review.

The ALMSS verifier uses $2N + M = O(S)$ random bits and makes 4 queries to the proof. The proof is expected to be a linear function $\pi \colon \{0,1\}^{N^2} \to \{0,1\}$, which can be represented via a vector of $N^2 = O(S^2)$ bits. The queries are input oblivious because they depend only on the language L and input size n, which in turn determine the circuit C.

The time complexity of the query sampler is dominated by the cost of constructing the matrices $P_1, \dots, P_M \in \{0,1\}^{N^2}$ and constants $c_1, \dots, c_M \in \{0,1\}$, which takes time $O(S^2)$.

The time complexity of the decision predicate is $O(n)$, because it takes $O(n)$ time to compute $\sum_{j=1}^{n} s_j c_j$, and the rest of the computation takes constant time. Note that the query sampler precomputes $\sum_{j=n+1}^{M} s_j c_j$, since it does not depend on x (only $c_1, \dots, c_n$ depend on x).

Completeness. Completeness of the ALMSS verifier is as in the classical setting. Namely, suppose that $C(x) = 1$, and let $\mathbf{w} = (w_1, \dots, w_N) \in \{0,1\}^N$ be such that w_i is the value of the i-th wire when C computes on x. Then, the linear function defined as $\pi(Z) := \langle \mathbf{w} \otimes \mathbf{w}, Z \rangle = \sum_{i,i' \in [N]} w_i w_{i'} \cdot Z_{i,i'}$ for all $Z \in \{0,1\}^{N^2}$ is accepted with probability 1. Indeed,

1. for every $u, v \in \{0,1\}^N$ it holds that

$$\pi(D_u)\pi(D_v) = \left(\sum_{i \in [N]} w_i u_i\right)\left(\sum_{i' \in [N]} w_{i'} v_{i'}\right) = \sum_{i,i' \in [N]} w_i w_{i'} u_i v_{i'} = \pi(u \otimes v) \ ;$$

2. for every $s \in \{0,1\}^M$ it holds that

$$\pi\Big(\sum_{j=1}^{M} s_j P_j\Big) = \sum_{j=1}^{M} s_j \pi(P_j) = \sum_{j=1}^{M} s_j c_j \ ,$$

because $\pi(P_j) = \langle \mathbf{w} \otimes \mathbf{w}, P_j \rangle = P_j(\mathbf{w}) = c_j$ for all $j \in [M]$.

Soundness. Soundness against non-signaling functions is quite unlike the classical case, and is discussed in the next few sub-sections. We shall prove that if the ALMSS verifier accepts a given linear k non-signaling function $\mathcal{L} \colon \{0,1\}^{N^2} \to \{0,1\}$ with sufficiently large constant probability, then $C(x) = 1$, and thus $x \in L$.

8.2 Step 1: Constructing the linear local assignment generator

We have proved in Section 7 that, to conclude that $C(x) = 1$, it suffices to construct a linear local assignment generator with sufficiently small error and sufficiently large locality. Here we show how to construct a linear local assignment generator from any linear k'-non-signaling function $\mathcal{L}'$ (possibly different from $\mathcal{L}$) that satisfies *every* test of the ALMSS verifier with large probability.

Lemma 8.1. *Let $k' \geq 4$. Suppose that $\mathcal{L}' \colon \{0,1\}^{N^2} \to \{0,1\}$ is a linear k'-non-signaling function for which the following holds.*

1. *Every tensor test is satisfied with probability $1 - \varepsilon_1$*

$$\forall\, x, y \in \{0,1\}^N,\ \Pr_{\mathcal{L}'}\left[\mathcal{L}'(D_x)\mathcal{L}'(D_y) = \mathcal{L}'(x \otimes y)\right] \geq 1 - \varepsilon_1 \ .$$

2. *Every satisfiability test is satisfied with probability $1 - \varepsilon_2$*

$$\forall\, s \in \{0,1\}^M,\ \Pr_{\mathcal{L}'}\left[\mathcal{L}'\left(\sum_{j=1}^{M} s_j P_j\right) = \sum_{j=1}^{M} s_j c_j\right] \geq 1 - \varepsilon_2 \ .$$

Then there exists a linear k'-local assignment generator $\mathcal{A}$ for (C, x) with error $\varepsilon' \leq \varepsilon_1 + \varepsilon_2$.

Proof. Being a linear k'-non-signaling function, $\mathcal{L}'$ can be represented uniquely as a quasi-distribution over linear functions $f\colon \{0,1\}^{N^2} \to \{0,1\}$. For each $B \in \{0,1\}^{N^2}$, let $\ell_B \in \mathbb{R}$ be the weight that $\mathcal{L}'$ assigns to the linear function $\langle B, \cdot \rangle$. Define $\mathcal{A}\colon \{0,1\}^N \to \{0,1\}$ to be the quasi-distribution over linear functions $f\colon \{0,1\}^N \to \{0,1\}$ that assigns to each linear function $\langle \alpha, \cdot \rangle$ the weight p_α, where

$$p_\alpha := \sum_{\substack{B \in \{0,1\}^{N^2} \\ \text{s.t. } \operatorname{diag}(B) = \alpha}} \ell_B \ ,$$

where $\operatorname{diag}(B) \in \{0,1\}^N$ is the vector on the diagonal of the matrix B. We now prove that $\mathcal{A}$ is a k'-local assignment generator for (C, x) with error $\varepsilon' \leq \varepsilon_1 + \varepsilon_2$.

Locality. The quasi-distribution $\mathcal{A}$ is k'-local because, for every subset $S \subseteq \{0,1\}^N$ of size at most k' and every string $\vec{b} \in \{0,1\}^S$,

$$\begin{aligned}
\widetilde{\Pr}\left[\mathcal{A}(S) = \vec{b}\right] &= \sum_{\substack{\alpha \in \{0,1\}^N \text{ s.t.} \\ \forall x \in S\ \langle \alpha, x \rangle = \vec{b}_x}} p_\alpha \\
&= \sum_{\substack{\alpha \in \{0,1\}^N \text{ s.t.} \\ \forall x \in S\ \langle \alpha, x \rangle = \vec{b}_x}} \left(\sum_{\substack{B \in \{0,1\}^{N^2} \\ \text{s.t. } \operatorname{diag}(B) = \alpha}} \ell_B \right) \\
&= \sum_{\substack{B \in \{0,1\}^{N^2} \\ \text{s.t. } \forall x \in S\ \langle \operatorname{diag}(B), x \rangle = \vec{b}_x}} \ell_B \\
&= \sum_{\substack{B \in \{0,1\}^{N^2} \\ \text{s.t. } \forall x \in S\ \langle B, D_x \rangle = \vec{b}_x}} \ell_B \\
&= \Pr\left[\forall x \in S\ \mathcal{L}'(D_x) = \vec{b}_x\right] \geq 0 \ . \quad (\text{as } \mathcal{L}' \text{ is } k'\text{-local})
\end{aligned}$$

Satisfying the constraints. The quasi-distribution $\mathcal{A}$ satisfies all M constraints if and only if for for every constraint P_j it holds that $P_j(\mathcal{A}(S_j)) = c_j$, where the set S_j contains the indices for the variables influencing P_j, and P_j is interpreted as the polynomial that accepts only these variables (i.e., ignoring the variables in the coordinates $[N] \setminus S_j$). Recall that each P_j depends on at most 3 variables. Therefore, we may query $\mathcal{A}$ on such sets S_j, provided that $\mathcal{A}$ is k'-local for $k' \geq 3$.

Below we show that for each $j \in [M]$ it holds that

$$\Pr[P_j(\mathcal{A}(S_j)) = c_j] \geq 1 - (\varepsilon_1 + \varepsilon_2) \ . \tag{2}$$

Suppose first that $P_j(\cdot) = c_j$ is a constraint that does not involve a quadratic term, so that it is either an input consistency constraint or a NOT gate constraint. We will prove the statement only for the case of an input consistency constraint; the case of a NOT gate is analogous, and we omit its proof. An input consistency constraint has the form $w_i = c_j$ for some $i \in [N]$ and $c_j \in \{0,1\}$. Therefore,

$$\Pr[P_j(\mathcal{A}(S_j)) = c_j] = \Pr[\mathcal{A}(e_i) = c_j] = \Pr[\mathcal{L}'(D_{e_i}) = c_j] = \Pr[\mathcal{L}'(P_j) = c_j] \geq 1 - \varepsilon_2 \ .$$

Otherwise, $P_j(\cdot) = c_j$ involves a quadratic term. More specifically, P_j either corresponds to an AND gate, and hence is of the form $w_i - w_{i_1} \cdot w_{i_2} = c_j$, or it corresponds to an OR gate, and is of the form $w_i - w_{i_1} - w_{i_2} + w_{i_1} \cdot w_{i_2} = c_j$. Below we prove that $\Pr[P_j(\mathcal{A}(S_j)) = c_j] \geq 1 - (\varepsilon_1 + \varepsilon_2)$ for the case of an AND gate; the case of the OR gate is analogous, and we omit its proof.

If P_j corresponds to an AND gate, then

$$\Pr[P_j(\mathcal{A}(S_j)) = c_j] = \Pr[\mathcal{A}(e_i) - \mathcal{A}(e_{i_1}) \cdot \mathcal{A}(e_{i_2}) = c_j] = \Pr[\mathcal{L}'(D_{e_i}) - \mathcal{L}'(D_{e_{i_1}}) \cdot \mathcal{L}'(D_{e_{i_2}}) = c_j]$$

Since $\mathcal{L}'$ individually satisfies each tensor test constraints with probability $1 - \varepsilon_1$ and $k' \geq 4$,

$$\Pr[\mathcal{L}'(D_{e_i}) - \mathcal{L}'(D_{e_{i_1}})\mathcal{L}'(D_{e_{i_2}}) = c_j] \geq \Pr[\mathcal{L}'(D_{e_i}) - \mathcal{L}'(e_{i_1} \otimes e_{i_2}) = c_j] - \varepsilon_1 \ .$$

We conclude that

$$\begin{aligned} \Pr[P_j(\mathcal{A}(S_j)) = c_j] &\geq \Pr[\mathcal{L}'(D_{e_i}) - \mathcal{L}'(e_{i_1} \otimes e_{i_2}) = c_j] - \varepsilon_1 \\ &= \Pr[\mathcal{L}'(D_{e_i} - e_{i_1} \otimes e_{i_2}) = c_j] - \varepsilon_1 \\ &= \Pr[\mathcal{L}'(P_j) = c_j] - \varepsilon_1 \\ &\geq 1 - \varepsilon_1 - \varepsilon_2 \ . \end{aligned}$$

□

8.3 Step 2: Relating the repeated verifier to the basic verifier

We argue that if a linear k-non-signaling function $\mathcal{L}$ is accepted by the ALMSS verifier with probability at least δ and $k = \Omega(t)$, then $\mathcal{L}$ is accepted by the t-repeated ALMSS verifier with probability at least $\gamma := \Omega((0.99\delta)^t)$. This step is generic (holds regardless of which verifier is considered), and follows by a simple application of Lemma 5.3 with $\tau := \frac{\delta}{100}$ and $r := \lceil \frac{400t}{\delta} \rceil$. Note that this requires $k \geq q \cdot r$, so it suffices to have $k \geq q \cdot \frac{4t}{\tau} + q = 4 \cdot \frac{400t}{\delta} + 4 = \Omega(t)$.

8.4 Step 3: Reducing the soundness error

We argue that if $\mathcal{L}$ passes the t-repeated ALMSS verifier with probability at least γ and $k = \Omega(t)$, then there exists a (possibly different) linear $(k - 4t)$-non-signaling function $\mathcal{L}'$ that satisfies *every* constraint of the (non-repeated) ALMSS verifier with high probability $(1 - 2^{-\Omega_\gamma(t)})$. Namely, if $\mathcal{L}$ satisfies the t-repeated ALMSS verifier *on average*, then $\mathcal{L}'$ satisfies the ALMSS verifier with high probability *in the worst case.* Informally, $\mathcal{L}'$ equals $\mathcal{L}$ conditioned on the t-repeated verifier passing.

Definition 8.2. *Given a linear k-non-signaling function $\mathcal{L}\colon \{0,1\}^{N^2} \to \{0,1\}$ and a parameter t, we let accept_t denote the event that the t-repeated ALMSS verifier accepts $\mathcal{L}$, namely, that for uniformly random $u^{(1)}, v^{(1)}, \ldots, u^{(t)}, v^{(t)} \in \{0,1\}^N$ and $s^{(1)}, \ldots, s^{(t)} \in \{0,1\}^M$ it holds that*

$$\forall\, i \in [t], \quad \mathcal{L}(D_{u^{(i)}})\mathcal{L}(D_{v^{(i)}}) = \mathcal{L}(u^{(i)} \otimes v^{(i)}) \quad \text{and} \quad \mathcal{L}\left(\sum_{j=1}^{M} s_j^{(i)} P_j\right) = \sum_{j=1}^{M} s_j^{(i)} c_j \ .$$

For $k' := k - 4t$, we define $\mathcal{L}'\colon \{0,1\}^{N^2} \to \{0,1\}$ be the linear k'-non-signaling function defined as

$$\Pr_{\mathcal{L}'}\left[\mathcal{L}'(S) = \vec{b}\right] := \Pr_{\mathcal{L}}\left[\mathcal{L}(S) = \vec{b} \mid \mathsf{accept}_t\right] = \frac{\Pr[\mathcal{L}(S) = \vec{b} \wedge \mathsf{accept}_t]}{\Pr[\mathsf{accept}_t]} \ .$$

Lemma 8.3. *For every $k, t \in \mathbb{N}$ such that $k \geq 51t$ and linear k-non-signaling function $\mathcal{L}\colon \{0,1\}^{N^2} \to \{0,1\}$, if $\Pr_{\mathcal{L}}[\mathsf{accept}_t] \geq \gamma$ then*

$$\forall\, x, y \in \{0,1\}^N, \quad \Pr_{\mathcal{L}'}\left[\mathcal{L}'(D_x)\mathcal{L}'(D_y) = \mathcal{L}'(x \otimes y)\right] \geq 1 - \varepsilon_1$$

$$\forall\, s \in \{0,1\}^M, \quad \Pr_{\mathcal{L}'}\left[\mathcal{L}'\left(\sum_{j=1}^{M} s_j P_j\right) = \sum_{j=1}^{M} s_j c_j\right] \geq 1 - \varepsilon_2 \ ,$$

where $\varepsilon_1, \varepsilon_2 \leq \frac{5}{\gamma}\left(\frac{4}{5}\right)^t$.

Proof. The proof relies on the following two claims.

Claim 8.4. *For every $d \in \mathbb{N}$ such that $k \geq \max\{12d + 3t, 4t + 3\}$,*

$$\forall\, x, y \in \{0,1\}^N, \quad \Pr_{\mathcal{L}'}\left[\mathcal{L}'(D_x)\mathcal{L}'(D_y) \neq \mathcal{L}'(x \otimes y)\right] \leq \frac{4}{\gamma}\left(\frac{d}{t+d}\right)^{\lfloor \frac{d-1}{4} \rfloor} \ .$$

Claim 8.5. *For every $d \in \mathbb{N}$ such that $k \geq \max\{t + 2d, 4t + 1\}$,*

$$\forall\, s \in \{0,1\}^M, \quad \Pr_{\mathcal{L}'}\left[\mathcal{L}'\left(\sum_{j=1}^{M} s_j P_j\right) \neq \sum_{j=1}^{M} s_j c_j\right] \leq \frac{2}{\gamma}\left(\frac{d}{t+d}\right)^{\lfloor d/2 \rfloor} \ .$$

We prove the claims further below, and for now show how they imply the lemma. By the assumption that $k \geq 51t$, we can set $d := 4t$ in order to get

$$\Pr_{\mathcal{L}'}[\mathcal{L}'(D_x)\mathcal{L}'(D_y) \neq \mathcal{L}'(x \otimes y)] \leq \frac{4}{\gamma}\left(\frac{4t}{5t}\right)^{\lfloor t - \frac{1}{4} \rfloor} \leq \frac{5}{\gamma}\left(\frac{4}{5}\right)^t \ ,$$

$$\Pr_{\mathcal{L}'}\left[\mathcal{L}'\left(\sum_{j=1}^{M} s_j P_j\right) \neq \sum_{j=1}^{M} s_j c_j\right] \leq \frac{2}{\gamma}\left(\frac{4t}{5t}\right)^{\lfloor 2t \rfloor} \leq \frac{2}{\gamma}\left(\frac{4}{5}\right)^{2t-1} \leq \frac{5}{\gamma}\left(\frac{4}{5}\right)^t \ . \qquad \square$$

We now prove the two claims.

Proof of Claim 8.4. Denote by tensor_t the event "$\mathcal{L}(D_{u^{(i)}})\mathcal{L}(D_{v^{(i)}}) = \mathcal{L}(u^{(i)} \otimes v^{(i)})$ for all $i \in [t]$". If we assume that x and y are chosen randomly, then a naive application of Lemma 5.1 implies that

$$\begin{aligned}\Pr_{\mathcal{L}'}[\mathcal{L}'(D_x)\mathcal{L}'(D_y) \neq \mathcal{L}'(x \otimes y)] &= \frac{\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{accept}_t]}{\Pr[\mathsf{accept}_t]} \\ &\leq \frac{\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t]}{\Pr[\mathsf{accept}_t]} \\ &\leq \frac{1}{\gamma(t+1)} \ .\end{aligned}$$

We will instead invoke the lemma more carefully and use the assumption that the locality of $\mathcal{L}$ is sufficiently large to prove a stronger bound for *fixed* x and y. Specifically, we use the assumption that $k \geq \max\{12d + 3t, 4t + 3\}$, together with the linearity of $\mathcal{L}$ to prove that for all $x, y \in \{0,1\}^N$, it holds that

$$\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t] \leq 4\left(\frac{d}{t+d}\right)^{\lfloor \frac{d-1}{4} \rfloor} \ . \qquad (3)$$

Indeed, this implies that

$$\begin{aligned}
\Pr_{\mathcal{L}'}[\mathcal{L}'(D_x)\mathcal{L}'(D_y) \neq \mathcal{L}'(x \otimes y)] &= \frac{\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{accept}_t]}{\Pr[\mathsf{accept}_t]} \\
&\leq \frac{\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t]}{\Pr[\mathsf{accept}_t]} \\
&\leq \frac{4}{\gamma}\left(\frac{d}{t+d}\right)^{\lfloor \frac{d-1}{4} \rfloor} ,
\end{aligned}$$

which is what we wanted to prove.

It remains to argue that Eq. (3) holds. Let $w_x^{(1)}, \ldots, w_x^{(d)}; w_y^{(1)}, \ldots, w_y^{(d)} \in \{0,1\}^n$ be $2d$ vectors chosen uniformly at random. By linearity of $\mathcal{L}$, for all $i \in [d]$ we have

$$\begin{aligned}
\mathcal{L}(D_x)\mathcal{L}(D_y) &= \mathcal{L}(D_{w_x^{(i)}})\mathcal{L}(D_{w_y^{(i)}}) + \mathcal{L}(D_{w_x^{(i)}})\mathcal{L}(D_{y+w_y^{(i)}}) \\
&\quad + \mathcal{L}(D_{x+w_x^{(i)}})\mathcal{L}(D_{w_y^{(i)}}) + \mathcal{L}(D_{x+w_x^{(i)}})\mathcal{L}(D_{y+w_y^{(i)}}) \ , \qquad (4) \\
\mathcal{L}(x \otimes y) &= \mathcal{L}(w_x^{(i)} \otimes w_y^{(i)}) + \mathcal{L}(w_x^{(i)} \otimes (y + w_y^{(i)})) \\
&\quad + \mathcal{L}((x + w_x^{(i)}) \otimes w_y^{(i)}) + \mathcal{L}((x + w_x^{(i)}) \otimes (y + w_y^{(i)})) \ . \qquad (5)
\end{aligned}$$

For a positive integer $r < d/4$, Lemma 5.1 implies that

$$\begin{aligned}
\Pr\left[\begin{matrix} \text{for at least } r \text{ indices } i \in [d] \\ \mathcal{L}(D_{w_x^{(i)}})\mathcal{L}(D_{w_y^{(i)}}) \neq \mathcal{L}(w_x^{(i)} \otimes w_y^{(i)}) \end{matrix} \wedge \mathsf{tensor}_t\right] &\leq \left(\frac{d}{t+d}\right)^r , \\
\Pr\left[\begin{matrix} \text{for at least } r \text{ indices } i \in [d] \\ \mathcal{L}(D_{w_x^{(i)}})\mathcal{L}(D_{y+w_y^{(i)}}) \neq \mathcal{L}(x \otimes (y + w_y^{(i)})) \end{matrix} \wedge \mathsf{tensor}_t\right] &\leq \left(\frac{d}{t+d}\right)^r , \\
\Pr\left[\begin{matrix} \text{for at least } r \text{ indices } i \in [d] \\ \mathcal{L}(D_{x+w_x^{(i)}})\mathcal{L}(D_y) \neq \mathcal{L}((x + w_x^{(i)}) \otimes w_y^{(i)}) \end{matrix} \wedge \mathsf{tensor}_t\right] &\leq \left(\frac{d}{t+d}\right)^r , \\
\Pr\left[\begin{matrix} \text{for at least } r \text{ indices } i \in [d] \\ \mathcal{L}(D_{x+w_x^{(i)}})\mathcal{L}(D_{y+w_y^{(i)}}) \neq \mathcal{L}((x + w_x^{(i)}) \otimes (y + w_y^{(i)})) \end{matrix} \wedge \mathsf{tensor}_t\right] &\leq \left(\frac{d}{t+d}\right)^r .
\end{aligned}$$

Denote by neq the union of the foregoing events. By union bound, $\Pr[\mathsf{neq}] \leq 4(\frac{d}{t+d})^r$, and hence

$$\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t] \leq \Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t \mid \neg\mathsf{neq}] + 4\left(\frac{d}{t+d}\right)^r .$$

We show next that $\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t \mid \neg\mathsf{neq}] = 0$. Indeed, if $\neg\mathsf{neq}$ holds and $4r < d$, then there exists $i \in [d]$ such that

- $\Pr[\mathcal{L}(D_{w_x^{(i)}})\mathcal{L}(D_{w_y^{(i)}}) = \mathcal{L}(w_x^{(i)} \otimes w_y^{(i)})$,
- $\Pr[\mathcal{L}(D_{w_x^{(i)}})\mathcal{L}(D_{y+w_y^{(i)}}) = \mathcal{L}(x \otimes (y + w_y^{(i)}))$,
- $\mathcal{L}(D_{x+w_x^{(i)}})\mathcal{L}(D_y) = \mathcal{L}((x + w_x^{(i)}) \otimes w_y^{(i)})$, and
- $\mathcal{L}(D_{x+w_x^{(i)}})\mathcal{L}(D_{y+w_y^{(i)}}) = \mathcal{L}((x + w_x^{(i)}) \otimes (y + w_y^{(i)}))$.

Therefore, by Eqs. (4) and (5), if $\neg\mathsf{neq}$ holds and $4r < d$ then $\mathcal{L}(x \otimes y) = \mathcal{L}(D_x)\mathcal{L}(D_y)$. This implies that $\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t \mid \neg\mathsf{neq}] = 0$, and so for $r = \lfloor \frac{d-1}{4} \rfloor$ we get that

$$\Pr[\mathcal{L}(D_x)\mathcal{L}(D_y) \neq \mathcal{L}(x \otimes y) \wedge \mathsf{tensor}_t] \leq 4\left(\frac{d}{t+d}\right)^r = 4\left(\frac{d}{t+d}\right)^{\lfloor \frac{d-1}{4} \rfloor} ,$$

which proves Eq. (3), and thus completes the proof of Claim 8.4. □

Proof of Claim 8.5. Let $s^{(1)}, \ldots, s^{(t)} \in \{0,1\}^M$ be chosen uniformly at random, and denote by sat_t the event that

$$\forall\, i \in [t] \quad \mathcal{L}\left(\sum_{j=1}^{M} s_j^{(i)} P_j\right) = \sum_{j=1}^{M} s_j^{(i)} c_j \ .$$

We use the assumption that $k \geq \max\{t + 2d, 4t + 1\}$ to prove that for all $s \in \{0,1\}^M$ it holds that

$$\Pr\left[\mathcal{L}\left(\sum_{j=1}^{M} s_j P_j\right) \neq \sum_{j=1}^{M} s_j c_j \wedge \mathsf{sat}_t\right] \leq 2\left(\frac{d}{t+d}\right)^{\lfloor d/2 \rfloor} . \tag{6}$$

Indeed, this implies that

$$\begin{aligned}\Pr\left[\mathcal{L}'\left(\sum_{j=1}^{M} s_j P_j\right) \neq \sum_{j=1}^{M} s_j c_j\right] &= \frac{\Pr\left[\mathcal{L}\left(\sum_{j=1}^{M} s_j P_j\right) \neq \sum_{j=1}^{M} s_j c_j \wedge \mathsf{accept}_t\right]}{\Pr[\mathsf{accept}_t]} \\ &\leq \frac{\Pr\left[\mathcal{L}\left(\sum_{j=1}^{M} s_j P_j\right) \neq \sum_{j=1}^{M} s_j c_j \wedge \mathsf{sat}_t\right]}{\Pr[\mathsf{accept}_t]} \\ &\leq \frac{2}{\gamma}\left(\frac{d}{t+d}\right)^{d/2} ,\end{aligned}$$

which is what we wanted to prove.

It remains to argue that Eq. (6) holds. Let $y^{(1)}, \ldots, y^{(d)} \in \{0,1\}^M$ be chosen uniformly at random. By linearity of $\mathcal{L}$ and Lemma 5.1, we have

$$\begin{aligned}&\Pr\left[\mathcal{L}\left(\sum_{j=1}^{M} s_j P_j\right) \neq \sum_{j=1}^{M} s_j c_j \wedge \mathsf{sat}_t\right] \\ &= \Pr_{\substack{s^{(1)},\ldots,s^{(t)} \in \{0,1\}^M \\ y^{(1)},\ldots,y^{(d)} \in \{0,1\}^M \\ \mathcal{L}}}\left[\begin{array}{c}\text{for every index } i \in [d] \\ \mathcal{L}\left(\sum_{j=1}^{M}(s_j + y_j^{(i)})P_j\right) + \mathcal{L}\left(\sum_{j=1}^{M} y_j^{(i)} P_j\right) \neq \sum_{j=1}^{M} s_j c_j\end{array} \wedge \mathsf{sat}_t\right] \\ &\leq \Pr\left[\begin{array}{c}\text{for at least } \lfloor d/2 \rfloor \text{ indices } i \in [d] \\ \mathcal{L}\left(\sum_{j=1}^{M}(s_j + y_j^{(i)})P_j\right) \neq \sum_{j=1}^{M}(s_j + y_j^{(i)})c_j\end{array} \wedge \mathsf{sat}_t\right] \\ &\quad + \Pr\left[\begin{array}{c}\text{for at least } \lfloor d/2 \rfloor \text{ indices } i \in [d] \\ \mathcal{L}\left(\sum_{j=1}^{M} y_j^{(i)} P_j\right) \neq \sum_{j=1}^{M} y_j^{(i)} c_j\end{array} \wedge \mathsf{sat}_t\right] \\ &\leq 2\left(\frac{d}{t+d}\right)^{\lfloor d/2 \rfloor} ,\end{aligned}$$

which proves Eq. (6), and thus completes the proof of Claim 8.5. □

8.5 Step 4: Constant soundness error of the ALMSS verifier

Suppose that $\mathcal{L}$ is a linear k-non-signaling function $\mathcal{L}$ that is accepted by the ALMSS verifier with probability at least $\delta = \frac{5}{6}$, where $k = O(\log N)$. We need to conclude that $C(x) = 1$.

Let $t := c \log_2 N$ for a sufficiently large constant c to be chosen later.

Since $k \geq 4 \cdot \frac{400t}{\delta} + 4$, the discussion in Section 8.3 implies that $\mathcal{L}$ is accepted by the t-repeated ALMSS verifier with probability at least $\gamma = \Omega((0.99\delta)^t)$.

By Lemma 8.3, there exists a linear k'-non-signaling function $\mathcal{L}'$ with $k' = k - 4t \geq 47t$ such that

$$\forall\, x, y \in \{0,1\}^N, \quad \Pr_{\mathcal{L}'}\left[\mathcal{L}'(D_x)\mathcal{L}'(D_y) = \mathcal{L}'(x \otimes y)\right] \geq 1 - \varepsilon_1$$

$$\forall\, s \in \{0,1\}^M, \quad \Pr_{\mathcal{L}'}\left[\mathcal{L}'\left(\sum_{j=1}^{M} s_j P_j\right) = \sum_{j=1}^{M} s_j c_j\right] \geq 1 - \varepsilon_2 \ ,$$

with $\varepsilon_1 + \varepsilon_2 \leq \frac{10}{\gamma}\left(\frac{4}{5}\right)^t = O(\left(\frac{4}{5} \cdot \frac{1}{0.99\delta}\right)^t) = O(\left(\frac{32}{33}\right)^t)$.

By applying Lemma 8.1 to $\mathcal{L}'$ we obtain a linear k'-local assignment generator $\mathcal{A}\colon \{0,1\}^N \to \{0,1\}$ for (C, x) with error $\varepsilon_1 + \varepsilon_2 = O(\left(\frac{32}{33}\right)^t)$.

By the choice of the parameters we have $t = c \log_2 N$ for a sufficiently large constant $c \in \mathbb{R}$ so that $O(\left(\frac{32}{33}\right)^t) < \frac{1}{40N^2}$ and $N^c > (40N^2)^2$.

Therefore, $\mathcal{A}$ is a linear k'-local assignment generator for (C, x) with error $\varepsilon' = \frac{1}{40N^2}$, and locality $k' \geq t = c \log_2 N \geq 2\log_2(40N^2) \geq 2 \log_2 \frac{1}{\varepsilon'}$. Moreover, $\varepsilon' \log_2(1/\varepsilon') \leq \frac{6 + 2\log_2 N}{40N^2} < \frac{1}{20N}$.

Hence, by applying Lemma 7.2 to $\mathcal{A}$, we get that $C(x) = 1$.

9 Testing linearity with low error

In this section we prove Theorem 7. Toward proving the theorem we define a *repeated self-correction* of $\mathcal{F}$, which we denote by $\overline{\mathcal{F}}_{t,d}$, and prove that if $\mathcal{F}$ passes the [BLR93] linearity test with probability $1-\varepsilon$, then (i) $\overline{\mathcal{F}}_{t,d}$ is close to some linear $\bar{k}$-non-signaling function $\mathcal{L}$, and (ii) for all for all query sets $S \subseteq \{0,1\}^n$ of size at most $\bar{k}$ it holds that the statistical distance between $\hat{\mathcal{F}}(S)$ and $\overline{\mathcal{F}}_{t,d}(S)$ is upper bounded by $O(|S|\sqrt{\varepsilon})$. We proceed with the details below.

9.1 Defining $\overline{\mathcal{F}}_{t,d}$

In order to define the repeated self-correction $\overline{\mathcal{F}}_{t,d}$ we will need the following two components, defined below.

1. Relaxed repeated versions of the [BLR93] linearity test parameterized by $t \in \mathbb{N}$ and $\rho \in [0,1)$. It is defined by the sampling procedure Q_t, and a predicate $\mathsf{D}_{t,\rho}$,
2. The encoding and decoding procedures Enc_d and Dec_d parameterized by $d \in \mathbb{N}$.

9.1.1 The repeated linearity test

The repeated linearity test parameterized by $t \in \mathbb{N}$ and $\rho \in [0,1]$ has two components: the query sampler, denoted by Q_t, and the decision predicate, denoted by $\mathsf{D}_{t,\rho}$.

Sampling queries: Q_t generates $3t$ queries by sampling $2t$ uniformly random vectors $x^{(1)}, y^{(1)}, \dots, x^{(t)}, y^{(t)} \in \{0,1\}^n$, and outputting $\left(x^{(i)}, y^{(i)}, x^{(i)} + y^{(i)}\right)_{i\in[t]}$. This is merely repeating the query procedure of the basic linearity test t times.

Decision predicate: $\mathsf{D}_{t,\rho}$ receives the $3t$ answers $(a_{x^{(i)}}, a_{y^{(i)}}, a_{x^{(i)}+y^{(i)}})_{i\in[t]}$ to the queries, and accepts if and only if $a_{x^{(i)}} + a_{y^{(i)}} = a_{x^{(i)}+y^{(i)}}$ for at least $(1-\rho)t$ indices $i \in [t]$. That is, the test accepts if at least $(1-\rho)$-fraction of the repetitions pass.

9.1.2 The encoding/decoding procedures

Encoding procedure: The definition of Enc_d is a bit more complicated, although the main idea is the same as in the classical setting. Given a set $S = (x_1, \dots, x_s) \subseteq \{0,1\}^n$ of size s, the encoding procedure Enc_d returns a collection of d pairs $\{(z_j^{(i)}, z_j^{(i)} + x_j) : i \in [d]\}$ for each $x_j \in S$ according to a particular distribution, which we describe below in three stages:

1. **A single query:** Suppose first that $S = \{x\}$ is has only one element. In this case $\mathsf{Enc}_d(S)$ samples $z^{(1)}, \dots, z^{(d)} \in \{0,1\}^n$ independently and uniformly at random, and returns
$$\mathsf{Enc}_d(\{x\}) = \{(z^{(i)}, z^{(i)} + x) : i \in [d]\} \enspace .$$
2. **Independent vectors:** Generalizing the previous example, suppose that $S = \{x_1, \dots, x_s\} \subseteq \{0,1\}^n$ is a linearly independent set. In this case $\mathsf{Enc}_d(S)$ samples $z_j^{(1)}, \dots, z_j^{(d)} \in \{0,1\}^n$ for each $x_j \in S$ independently and uniformly at random, and returns
$$\mathsf{Enc}_d(S) = \{(z_j^{(i)}, z_j^{(i)} + x_j) : i \in [d]\}_{x_j \in S} \enspace .$$

3. **General case:** In the case of general $S = \{x_1, \dots, x_s\}$, Enc_d is essentially the same, only now for every $i \in [d]$ we require that the $z_j^{(i)}$'s satisfy the same linear relations as the x_j's. Formally, we define the following distribution $\mathcal{Z}^d(S)$.

   Definition 9.1. *For a set $S = (x_1, \dots, x_s) \subseteq \{0,1\}^n$ and a parameter $d \in \mathbb{N}$, we define the distribution $\mathcal{Z}^d(S)$ that outputs $\{z_j^{(i)} \in \{0,1\}^n\}_{x_j \in S, i \in [d]}$ as follows. Let $\beta_1, \dots \beta_r$ be a basis for* $\operatorname{span}(S)$*, so that for every $j \in [s]$, $x_j = \sum_{l=1}^{r} c_{j,l}\beta_l$ for some $c_{j,l} \in \{0,1\}$. Sample $w_j^{(i)} \in \{0,1\}^n$ independently and uniformly at random for all $i \in [d], j \in [r]$. The output of $\mathcal{Z}^d(S)$ is $\{z_j^{(i)}\}_{x_j \in S, i \in [d]}$, where $z_j^{(i)} = \sum_{l=1}^{r} c_{j,l} w_l^{(i)}$.*

   We note that $\mathcal{Z}^d(S)$ does not depend on the choice of basis $\beta_1, \dots \beta_r$.

   Using the above definition, we define Enc_d for general S as follows. Enc_d samples $\{z_j^{(i)}\}_{x_j \in S, i \in [d]}$ according to $\mathcal{Z}^d(S)$ and returns

$$\mathsf{Enc}_d(S) = \{(z_j^{(i)}, z_j^{(i)} + x_j) : i \in [d]\}_{x_j \in S} \ .$$

This completes the specification of Enc_d.

Decoding procedure: The decoding procedure Dec_d is the same as in the classical setting. It gets as input $\mathcal{F}(\mathsf{Enc}_d(S))$, the answers of $\mathcal{F}$ to the encoding S, and returns the vector $\vec{b} = (b_1, \dots, b_s) \in \{0,1\}^S$ of answers defined as

$$b_j = \operatorname*{maj}_{i \in [d]} \left\{ \mathcal{F}(z_j^{(i)} + x_j) - \mathcal{F}(z_j^{(i)}) \right\} \ .$$

9.1.3 The definition of $\overline{\mathcal{F}}_{t,d}$

Given the foregoing components, the conditioned self-correction $\overline{\mathcal{F}}_{t,d}$ of $\mathcal{F}$ is defined as

$$\Pr[\overline{\mathcal{F}}_{t,d}(S) = \vec{b}] := \Pr[\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}_d(S))) = \vec{b} \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1]$$

for every $S \subseteq \{0,1\}^n$ of some maximal size and $\vec{b} \in \{0,1\}^S$. That is, on an input x, the self-correction $\overline{\mathcal{F}}_{t,d}$ outputs the majority over d corrections, conditioned on the event that $(1-\rho)$-fraction of the t repetitions of the basic linearity test pass.

9.2 $\overline{\mathcal{F}}_{t,d}$ is non-signaling

In this section, we prove the following lemma.

Lemma 9.2. *If $\mathcal{F}$ is k-non-signaling, then $\overline{\mathcal{F}}_{t,d}$ is a $\bar{k}$-non-signaling function, provided that $k \geq 2d\bar{k} + 3t$.*

Proof. We have defined Enc_d as a function of a distribution $\mathcal{Z}^d(S)$ that outputs $(z_x^{(i)} \in \{0,1\}^n)_{x \in S, i \in [d]}$, where $\mathcal{Z}^d(S)$ depends on S. In the following claim, we show that there is a global distribution $\mathcal{Y}^d$ (independent of S) that outputs $(y_x^{(i)} \in \{0,1\}^n)_{x \in \{0,1\}^n, i \in [d]}$, such that for each $S \subseteq \{0,1\}^n$ the distribution $\mathcal{Z}^d(S)$ is equal to $\mathcal{Y}^d|_S$, the restriction of $\mathcal{Y}^d$ to S.

Claim 9.3. *There exists a global distribution $\mathcal{Y}^d$ over $(y_x^{(i)} \in \{0,1\}^n)_{x \in \{0,1\}^n, i \in [d]}$ such that $\mathcal{Y}^d|_S$ is equal to $\mathcal{Z}^d(S)$ for all sets $S \subseteq \{0,1\}^n$.*

Given Claim 9.3, the rest of the proof is rather straightforward. Specifically, we prove that for all $S' \supseteq S$ and $\vec{b} \in \{0,1\}^S$ it holds that $\Pr[\overline{\mathcal{F}}_{t,d}(S')|_S = \vec{b}] = \Pr[\overline{\mathcal{F}}_{t,d}(S) = \vec{b}]$. Define $\mathsf{Enc}'_d(S)$ to be the same algorithm as $\mathsf{Enc}_d(S)$, only it uses $\mathcal{Y}^d|_S$ instead of $\mathcal{Z}^d(S)$. Given $S' \supseteq S$ and $\vec{b} \in \{0,1\}^S$, by Claim 9.3 we have

$$\begin{aligned}\Pr[\overline{\mathcal{F}}_{t,d}(S')|_S = \vec{b}] &= \Pr\left[\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}_d(S')))|_S = \vec{b} \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \\ &= \Pr\left[\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}'_d(S')))|_S = \vec{b} \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \ .\end{aligned}$$

The specification of Dec_d and the non-signaling property of $\mathcal{F}$ imply that $\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}'_d(S')))|_S$ is equal to $\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}'_d(S)))$. Indeed, Dec_d decodes each element separately, and so, by the non-signaling property of $\mathcal{F}$ and the fact that $\mathcal{Y}$ is independent of S, encoding S' with Enc'_d and then decoding only the elements of S (and ignoring the elements in $S' \setminus S$) is equivalent to encoding S with Enc'_d and then decoding. Hence,

$$\begin{aligned}&\Pr\left[\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}'_d(S')))|_S = \vec{b} \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \\ &= \Pr\left[\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}'_d(S))) = \vec{b} \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \\ &= \Pr\left[\mathsf{Dec}_d(\mathcal{F}(\mathsf{Enc}_d(S))) = \vec{b} \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \\ &= \Pr[\overline{\mathcal{F}}_{t,d}(S) = \vec{b}] \ ,\end{aligned}$$

and so $\overline{\mathcal{F}}_{t,d}$ is non-signaling. Note that all the probabilities are well-defined provided that $k \geq 2d\,|S| + 3t$, which holds for $|S| \leq \bar{k}$. This completes the proof of Lemma 9.2. □

We now prove Claim 9.3.

Proof of Claim 9.3. We first define the distribution $\mathcal{Y}^d$ supported on $\{(y_x^{(i)})_{i \in [d]} : x \in \{0,1\}^n\}$, and then claim that its marginal distribution restricted to S coincides with $\mathcal{Z}^d(S)$ for all $S \subseteq \{0,1\}^n$.

The distribution $\mathcal{Y}^d$ is defined by sampling nd independent and uniformly random vectors $w_j^{(i)} \in \{0,1\}^n$ for each $j \in [n]$ and $i \in [d]$, and letting $y_x^{(i)} = \sum_{j=1}^n a_j \cdot w_j^{(i)} \in \{0,1\}^n$ for all $x = (a_1, \ldots, a_n) \in \{0,1\}^n$. The distribution $\mathcal{Y}^d$ outputs $(y_x^{(i)})_{x \in \{0,1\}^n, i \in [d]}$.

Next, we show that for each $S \subseteq \{0,1\}^n$ the distribution $\mathcal{Z}^d(S)$ is equal to $\mathcal{Y}^d|_S$. Indeed, Claim 9.3 follows from the following two claims.

Claim 9.4. *Fix a set $S = \{x_1, \ldots, x_s\} \subseteq \{0,1\}^n$. Suppose that $r = \dim \operatorname{span}(S)$, the vectors $x_1, \ldots, x_r \in S$ are linearly independent, and $x_j = c_{j,1}x_1 + \cdots + c_{j,r}x_r$ for all $j > r$. Then in the sampling of $\mathcal{Y}^d|_S$, the following holds.*

1. *The vectors $(y_j^{(i)})_{j \in [r], i \in [d]}$ are rd independent uniformly distributed vectors in $\{0,1\}^n$.*

2. *For all $j > r$ it holds that $y_j^{(i)} = c_{j,1}y_1^{(i)} + \cdots + c_{j,r}y_r^{(i)}$.*

Claim 9.5. *Fix a set $S = \{x_1, \ldots, x_s\} \subseteq \{0,1\}^n$. Suppose that $r = \dim \operatorname{span}(S)$, the vectors $x_1, \ldots, x_r \in S$ are linearly independent, and $x_j = c_{j,1}x_1 + \cdots + c_{j,r}x_r$ for all $j > r$. Then in the sampling of $\mathcal{Z}^d(S)$, for any choice of the basis $\beta_1, \ldots, \beta_r \in \{0,1\}^n$ the following holds.*

1. The vectors $(z_j^{(i)})_{j\in[r],i\in[d]}$ are rd independent uniformly distributed vectors in $\{0,1\}^n$.

2. For all $j > r$ it holds that $z_j^{(i)} = c_{j,1}z_1^{(i)} + \cdots + c_{j,r}z_r^{(i)}$.

The two foregoing claims clearly imply Claim 9.3. We now turn to the proofs of Claim 9.4 and Claim 9.5.

Proof of Claim 9.4. By definition of $\mathcal{Y}^d$, for each $i \in [d]$ we can write

$$\begin{bmatrix} | & & | & | & & | \\ y_1^{(i)} & \dots & y_r^{(i)} & y_{r+1}^{(i)} & \dots & y_s^{(i)} \\ | & & | & | & & | \end{bmatrix} = \begin{bmatrix} | & & | \\ w_1^{(i)} & \dots & w_n^{(i)} \\ | & & | \end{bmatrix} \cdot \begin{bmatrix} | & & | & | & & | \\ x_1 & \dots & x_r & x_{r+1} & \dots & x_s \\ | & & | & | & & | \end{bmatrix} \ ,$$

where the $w_j^{(i)}$'s are uniform and mutually independent. Therefore, since the set $\{x_j : j \in [r]\}$ is linearly independent, it follows that $\{y_j^{(i)} : j \in [r]\}$ are uniform and mutually independent. The second part of the claim is immediate by the assumption that $x_j = c_{j,1}x_1 + \cdots + c_{j,r}x_r$ for all $j > r$. □

Proof of Claim 9.5. Fix a set $S = \{x_1, \dots, x_s\} \subseteq \{0,1\}^n$, where $r = \dim \mathrm{span}(S)$ and $x_1, \dots, x_r$ are linearly independent vectors, and let $\beta_1, \dots, \beta_r \in \{0,1\}^n$ be a basis for $\mathrm{span}(S)$.

We need to prove that $(z_j^{(i)})_{j\in[r],i\in[d]}$ are rd independent uniformly distributed vectors in $\{0,1\}^n$. Denote by $M \in \{0,1\}^{r\times r}$ the binary matrix such that

$$\begin{bmatrix} | & & | \\ x_1 & \dots & x_r \\ | & & | \end{bmatrix} = \begin{bmatrix} | & & | \\ \beta_1 & \dots & \beta_r \\ | & & | \end{bmatrix} \cdot M \ .$$

Indeed, such M exists since the j-th column of M contains exactly the coefficients that allow us to represent x_j as a linear combination of $\beta_1, \dots, \beta_r$, namely, $x_j = M_{1,j}\beta_1 + \cdots + M_{r,j}\beta_r$. Then, given uniformly random $\{w_1^{(i)}, \dots, w_r^{(i)} : i \in [d]\}$, the vectors $z_1^{(i)}, \dots, z_r^{(i)}$ outputted by $\mathcal{Z}^d(S)$ are

$$\begin{bmatrix} | & & | \\ z_1^{(i)} & \dots & z_r^{(i)} \\ | & & | \end{bmatrix} = \begin{bmatrix} | & & | \\ w_1^{(i)} & \dots & w_r^{(i)} \\ | & & | \end{bmatrix} \cdot M$$

for each $i \in [d]$ independently. Therefore, since $w_1^{(i)} \dots w_r^{(i)}$ have been chosen independently and uniformly at random, and M is invertible we conclude that $(z_j^{(i)})_{j\in[r],i\in[d]}$ are rd vectors distributed in $\{0,1\}^n$ independently and uniformly. The second part of the claim is immediate by the assumption that $x_j = c_{j,1}x_1 + \cdots + c_{j,r}x_r$ for all $j > r$ and the definition of $\mathcal{Z}^d(S)$. □

Claim 9.3 follows immediately from Claim 9.4 and Claim 9.5, so we are done. □

9.3 $\overline{\mathcal{F}}_{t,d}$ is close to linear

In this section we show that if $\Pr[\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1]$ is bounded away from zero, then $\overline{\mathcal{F}}_{t,d}$ is close to linear assuming that t and d are sufficiently large. Specifically, we show prove the following lemma.

Lemma 9.6. *For any $\varepsilon > 0$ and $\gamma > 0$, if $t = d = O(\bar{k} + \log\frac{1}{\varepsilon} + \log\frac{1}{\gamma})$ and $\Pr[\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq \gamma$ and $\rho \leq 1/20$, then there exists a linear $\bar{k}$-non-signaling function $\mathcal{L}$ such that $\Delta_{\bar{k}}(\mathcal{L}, \overline{\mathcal{F}}_{t,d}) < \varepsilon$.*

Proof. The proof relies on the following theorem on linearity testing, due to [CMS18].

Theorem 8 ([CMS18])**.** *Let $\hat{\mathcal{F}}$ be a $\bar{k}$-non-signaling function such that for all $x, y \in \{0,1\}^n$, $\Pr[\hat{\mathcal{F}}(x) + \hat{\mathcal{F}}(y) = \hat{\mathcal{F}}(x+y)] \geq 1 - \hat{\varepsilon}$ and $\Pr[\hat{\mathcal{F}}(0^n) = 0] = 1$. Then there exists a linear $\bar{k}$-non-signaling function $\mathcal{L}$ such that $\Delta_{\bar{k}}(\hat{\mathcal{F}}, \mathcal{L}) \leq 2^{2\bar{k}}\hat{\varepsilon}$.*

The key step of the proof of Lemma 9.6 will be to prove the following lemma.

Lemma 9.7. *Let $\overline{\mathcal{F}}_{t,d}$ be the correction of $\mathcal{F}$, and suppose that $\Pr[\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq \gamma$. If $\rho < 1/16$, then for all $x, y \in \{0,1\}^n$ it holds that*

$$\Pr[\overline{\mathcal{F}}_{t,d}(x) + \overline{\mathcal{F}}_{t,d}(y) = \overline{\mathcal{F}}_{t,d}(x+y)] \geq 1 - \hat{\varepsilon} \tag{7}$$

for $\hat{\varepsilon} = \frac{6}{\gamma} \cdot e^{-\frac{(1/16-\rho)^2 \cdot \min\{t,d\}}{8}} + \frac{c' \cdot d \cdot 2^{-d/16}}{\gamma} + \frac{2}{\gamma} \cdot e^{-\frac{(1/8-\rho)^2 \cdot \min\{t,d\}}{8}}$, where c' is some absolute constant.

Indeed, by Theorem 8 this implies that there exists a linear $\bar{k}$-non-signaling function $\mathcal{L}\colon \{0,1\}^n \to \{0,1\}$ such that

$$\Delta_{\bar{k}}\left(\overline{\mathcal{F}}_{t,d}, \mathcal{L}\right) \leq 2^{2\bar{k}} \cdot \hat{\varepsilon} \ .$$

Therefore, for $\rho \leq 1/20$ and $t = d = O(\bar{k} + \log(1/\varepsilon) + \log(1/\gamma))$ we get that

$$\Delta_{\bar{k}}\left(\overline{\mathcal{F}}_{t,d}, \mathcal{L}\right) \leq 2^{2\bar{k}} \cdot \left(\frac{6}{\gamma} \cdot e^{-\Omega(t)} + \frac{c' \cdot t \cdot 2^{-\Omega(t)}}{\gamma} + \frac{2}{\gamma} \cdot e^{-\Omega(t)}\right) < \varepsilon \ ,$$

which proves Lemma 9.6. □

We now prove Lemma 9.7. Recall that $\overline{\mathcal{F}}_{t,d}$ is defined as $\mathsf{Dec}_d \circ \mathcal{F} \circ \mathsf{Enc}_d$ conditioned on $\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1$. For the query set $S = \{x, y, x+y\}$, the encoding $\mathsf{Enc}_d(S)$ can be split into three sets: $\{(z_x^{(i)}, z_x^{(i)} + x) : i \in [d]\}$, $\{(z_y^{(i)}, z_y^{(i)} + y) : i \in [d]\}$, and $\{(z_{x+y}^{(i)}, z_{x+y}^{(i)} + x + y) : i \in [d]\}$, where $z_x^{(1)}, \dots, z_x^{(d)}; z_y^{(1)}, \dots, z_y^{(d)} \in \{0,1\}^n$ are independent and uniform in $\{0,1\}^n$, and $z_{x+y}^{(i)} = z_x^{(i)} + z_y^{(i)}$ for all $i \in [d]$.

It will be convenient to define $b_x^{(i)} = \mathcal{F}(z_x^{(i)} + x) - \mathcal{F}(z_x^{(i)})$ for all $i \in [d]$. Similarly, define $b_y^{(i)} = \mathcal{F}(z_y^{(i)} + y) - \mathcal{F}(z_y^{(i)})$ and $b_{x+y}^{(i)} = \mathcal{F}(z_{x+y}^{(i)} + x + y) - \mathcal{F}(z_{x+y}^{(i)})$ for all $i \in [d]$. Using this notation, the decoding Dec_d returns the answer $\mathrm{maj}_{i \in [d]}\left\{b_x^{(i)}\right\}$ for x. Analogously, the decoding Dec_d returns $\mathrm{maj}_{i \in [d]}\left\{b_y^{(i)}\right\}$ for y, and $\mathsf{Dec}_d(x+y) = \mathrm{maj}_{i \in [d]}\left\{b_{x+y}^{(i)}\right\}$ for $x+y$.

The proof of Lemma 9.7 consists of the following 3 steps.

1. We first show that, if we condition on $\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1$, then with high probability there is some $b_x \in \{0,1\}$ such that $b_x^{(i)} = b_x$ for most indices $i \in [d]$. By symmetry, this also holds for y and $x+y$.

2. We then show that, if we condition on $\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1$, then with high probability $b_x^{(i)} + b_y^{(i)} = b_{x+y}^{(i)}$ for most indices $i \in [d]$.

3. Finally, by union bound all these events happen simultaneously with high probability, implying that $\overline{\mathcal{F}}_{t,d}(x) + \overline{\mathcal{F}}_{t,d}(y) = \overline{\mathcal{F}}_{t,d}(x+y)$ with high probability.

9.3.1 Step 1: Majority is a vast majority

Lemma 9.8. *Let $\rho \in [0,1]$ and $t,d,r \in \mathbb{N}$ be parameters such that d is even, $r < d/2$, and $\rho < r/4d$. Let $\mathcal{F}$ be a k-non-signaling function with $k \geq 6d+3t$. Fix $x \in \{0,1\}^n$, and let $x^{(1)},\dots,x^{(d)} \in \{0,1\}^n$ be independent uniformly random vectors. Denote by $\mathrm{Agr}_q(x)$ the event that $\exists b \in \{0,1\}$ such that $\mathcal{F}(x^{(j)}+x) - \mathcal{F}(x^{(j)}) = b$ for at least q indices $j \in [d]$. Then*

$$\Pr[\mathrm{Agr}_{d-r}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq 1 - \frac{2}{\gamma} \cdot e^{-\frac{(r/4d-\rho)^2 \cdot \min\{t,d\}}{8}} - \frac{c \cdot d \cdot 2^{-r/4}}{\gamma} \ ,$$

where c is some absolute constant.

Proof. Instead of arguing about $\mathrm{Agr}_q(x)$ directly, we first define a related event $\mathrm{PairAgr}_{q'}(x)$ for some q' that depends on q, and show that $\Pr[\mathrm{PairAgr}_{q'}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1]$ is high. We then relate $\mathrm{PairAgr}_{q'}(x)$ to $\mathrm{Agr}_q(x)$.

Denote by $\mathrm{PairAgr}_q(x)$ the event that $\mathcal{F}(x^{(2j-1)}+x)-\mathcal{F}(x^{(2j-1)}) = \mathcal{F}(x^{(2j)}+x)-\mathcal{F}(x^{(2j)})$ for at least q indices $j \in [d/2]$. Note that if $\mathcal{F}(x^{(2j-1)}+x)-\mathcal{F}(x^{(2j-1)}) \neq \mathcal{F}(x^{(2j)}+x)-\mathcal{F}(x^{(2j)})$, then either $\mathcal{F}(x^{(2j-1)}+x)+\mathcal{F}(x^{(2j)}) \neq \mathcal{F}(x^{(2j-1)}+x^{(2j)}+x)$ or $\mathcal{F}(x^{(2j)}+x)+\mathcal{F}(x^{(2j-1)}) \neq \mathcal{F}(x^{(2j-1)}+x^{(2j)}+x)$. Therefore,

$$\begin{aligned}
&\Pr\left[\neg\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \\
&\leq \Pr\left[\begin{array}{c}\text{for} \geq r/8 \text{ indices } j \in [d/2] \\ \mathcal{F}(x^{(2j-1)}+x)+\mathcal{F}(x^{(2j)}) \neq \mathcal{F}(x^{(2j-1)}+x^{(2j)}+x)\end{array} \middle| \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \\
&\quad + \Pr\left[\begin{array}{c}\text{for} \geq r/8 \text{ indices } j \in [d/2] \\ \mathcal{F}(x^{(2j)}+x)+\mathcal{F}(x^{(2j-1)}) \neq \mathcal{F}(x^{(2j-1)}+x^{(2j)}+x)\end{array} \middle| \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \ .
\end{aligned}$$

Since $x^{(2j-1)}+x$ and $x^{(2j)}$ are uniformly random and independent, by Lemma 5.2 each of the two terms is upper bounded by $\frac{1}{\gamma} \cdot e^{-\frac{(r/4d-\rho)^2 \cdot \min\{t,d\}}{8}}$. Therefore,

$$\Pr\left[\neg\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \leq \frac{2}{\gamma} \cdot e^{-\frac{(r/4d-\rho)^2 \cdot \min\{t,d\}}{8}} \ .$$

In the next claim we relate $\mathrm{PairAgr}_{q'}(x)$ to $\mathrm{Agr}_q(x)$.

Claim 9.9. *Let $d,r \in \mathbb{N}$ be parameters such that d is even and $r < d/2$. There exists an absolute constant $c > 0$ (independent of d or r) such that $\Pr[\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) \mid \neg\mathrm{Agr}_{d-r}(x)] \leq c \cdot d \cdot 2^{-r/4}$.*

Proof. For each $i \in [d]$ let $b_i = \mathcal{F}(x^{(i)}+x) - \mathcal{F}(x^{(i)}) \in \{0,1\}$, let $b^* = \mathrm{maj}\{b_i\}$ and let $q = |\{i \in [d] : b_i = b^*\}|$. Then $\mathrm{Agr}_{d-r}(x)$ holds if and only if $q \geq d-r$. We stress that $\mathrm{Agr}_{d-r}(x)$ is independent of the order of the indices $i \in [d]$. Next, we use the following lemma to relate $\mathrm{PairAgr}_{q'}(x)$ to $\mathrm{Agr}_q(x)$.

Lemma 9.10. *Let d be an even integer, and let $b_1,\dots,b_d \in \{0,1\}$ be such that not all b_i's are equal. Let $q = |\{i \in [d] : b_i = 0\}|$, and let $\alpha = \min\{\frac{q}{d}, 1-\frac{q}{d}\}$ for some $\alpha \in (0,0.5]$. If $\sigma\colon [d] \to [d]$ is a uniformly random permutation of $[d]$, then*

$$\Pr_{\sigma}[\textit{there are at least } \alpha d/4 \textit{ indices } i \in [d/2] \textit{ such that } b_{\sigma(2i-1)} \neq b_{\sigma(2i)}] \geq 1 - c \cdot d \cdot 2^{-\alpha d/4}$$

for some absolute constant $c > 0$.

We postpone the proof of lemma to Appendix A, and show now how it implies the claim. Indeed, by conditioning on $\neg\mathrm{Agr}_{d-r}(x)$, or equivalently on $d/2 \leq q < d-r$, and choosing a random permutation of the indices $i \in [d]$, by Lemma 9.10 we have

$$\Pr[\exists \text{ at least } r/4 \text{ indices } i \in [d/2] \text{ such that } b_{\sigma(2i-1)} \neq b_{\sigma(2i)} \mid \neg\mathrm{Agr}_{d-r}(x)] \geq 1 - c \cdot d \cdot 2^{-r/4} \enspace ,$$

for some absolute constant $c > 0$, independent of d or r. Therefore, $\Pr[\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) | \neg\mathrm{Agr}_{d-r}(x)] \leq c \cdot d \cdot 2^{-r/4}$. □

We use Claim 9.9 to complete the proof of Lemma 9.8. We have that

$$\begin{aligned}
1 - \frac{2}{\gamma} \cdot e^{-\frac{(r/4d-\rho)^2 \cdot \min\{t,d\}}{8}} &\leq \Pr[\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \\
&\leq \Pr[\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) \wedge \mathrm{Agr}_{d-r}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \\
&+ \Pr[\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) \wedge \neg\mathrm{Agr}_{d-r}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \\
&\leq \Pr[\mathrm{Agr}_{d-r}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] + \frac{\Pr[\mathrm{PairAgr}_{\frac{d}{2}-\frac{r}{4}}(x) \mid \neg\mathrm{Agr}_{d-r}(x)]}{\Pr[\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1]} \enspace .
\end{aligned}$$

Therefore,

$$\Pr[\mathrm{Agr}_{d-r}(x) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq 1 - \frac{2}{\gamma} \cdot e^{-\frac{(r/4d-\rho)^2 \cdot \min\{t,d\}}{8}} - \frac{c \cdot d \cdot 2^{-r/4}}{\gamma}$$

for some absolute constant $c \in \mathbb{R}$. This concludes the proof of Lemma 9.8. □

9.3.2 Step 2: Individual indices behave linearly

In step 2, we show that with high probability, $b_x^{(i)} + b_y^{(i)} = b_{x+y}^{(i)}$ for most indices $i \in [d]$.

Lemma 9.11. *Let $t, d, r \in \mathbb{N}$ and $\rho \in [0,1]$ be parameters such that $1 \leq r \leq d/2$ and $\rho < r/d$. Let $\mathcal{F}$ be a k-non-signaling function with $k \geq 6d+3t$. Fix $x, y \in \{0,1\}^n$, and let $z_x^{(1)}, \dots, z_x^{(d)}; z_y^{(1)}, \dots, z_y^{(d)} \in \{0,1\}^n$ be independent and uniformly distributed vectors in $\{0,1\}^n$. Define $b_x^{(i)} = \mathcal{F}(z_x^{(i)} + x) - \mathcal{F}(z_x^{(i)})$ for all $i \in [d]$, $b_y^{(i)} = \mathcal{F}(z_y^{(i)} + y) - \mathcal{F}(z_y^{(i)})$, and $b_{x+y}^{(i)} = \mathcal{F}(z_{x+y}^{(i)} + x + y) - \mathcal{F}(z_{x+y}^{(i)})$ for all $i \in [d]$. Then*

$$\Pr\left[\begin{array}{c} \exists \textit{ more than } d-2r \textit{ indices } i \in [d] \\ \textit{s.t. } b_x^{(i)} + b_y^{(i)} = b_{x+y}^{(i)} \end{array} \middle| \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1 \right] \geq 1 - \frac{2}{\gamma} \cdot e^{-\frac{(r/d-\rho)^2 \cdot \min\{t,d\}}{8}} \enspace ,$$

where the probability is over the random $z_x^{(i)}$'s,$z_y^{(i)}$'s and the randomness of $\mathcal{F}$.

Proof. By Lemma 5.2, for uniformly random $z_x^{(i)}, z_y^{(i)}$, and $z_{x+y}^{(i)} = z_x^{(i)} + z_y^{(i)}$ it holds that

$$\Pr\left[\begin{array}{c} \text{for at least } d-r+1 \text{ indices } i \in [d] \\ \mathcal{F}(z_x^{(i)}) + \mathcal{F}(z_y^{(i)}) = \mathcal{F}(z_{x+y}^{(i)}) \end{array} \middle| \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1 \right] \geq 1 - \frac{1}{\gamma} e^{-\frac{(r/d-\rho)^2 \cdot \min\{t,d\}}{8}} \enspace .$$

Similarly, we have

$$\Pr\left[\begin{array}{c} \text{for at least } d-r+1 \text{ indices } i \in [d] \\ \mathcal{F}(z_x^{(i)} + x) + \mathcal{F}(z_y^{(i)} + y) = \mathcal{F}(z_{x+y}^{(i)} + x + y) \end{array} \middle| \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1 \right] \geq 1 - \frac{1}{\gamma} e^{-\frac{(r/d-\rho)^2 \cdot \min\{t,d\}}{8}} \enspace .$$

Lemma 9.11 follows immediately by the union bound over the two events. □

9.3.3 Step 3: Putting it together

Finally, we combine the results in Lemma 9.8 and Lemma 9.11 in order to prove Lemma 9.7.

Proof of Lemma 9.7. Let $z_x^{(1)},\dots,z_x^{(d)}; z_y^{(1)},\dots,z_y^{(d)} \in \{0,1\}^n$ uniformly random vectors in $\{0,1\}^n$, and let $z_{x+y}^{(i)} = z_x^{(i)} + z_y^{(i)}$ for all $i \in [d]$, sampled by Enc_d. Define $b_x^{(i)} = \mathcal{F}(z_x^{(i)} + x) - \mathcal{F}(z_x^{(i)})$ for all $i \in [d]$. Similarly, define $b_y^{(i)} = \mathcal{F}(z_y^{(i)} + y) - \mathcal{F}(z_y^{(i)})$ and $b_{x+y}^{(i)} = \mathcal{F}(z_{x+y}^{(i)} + (x+y)) - \mathcal{F}(z_{x+y}^{(i)})$ for all $i \in [d]$. Applying Lemma 9.8 with $r = \frac{d}{4}$, we get that for some absolute constant $c > 0$,

$$\Pr\left[\exists b_x \in \{0,1\} \text{ s.t. } \left|\{i \in [d] : b_x^{(i)} = b_x\}\right| \geq \frac{3d}{4} \,\middle|\, \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \geq 1 - \varepsilon$$

for $\varepsilon = \frac{2}{\gamma}\cdot e^{-\frac{(1/16-\rho)^2\cdot\min\{t,d\}}{8}} + \frac{c\cdot d\cdot 2^{-d/16}}{\gamma}$. Note that if the foregoing event holds, then Dec_d returns b_x for x, and so $\overline{\mathcal{F}}_{t,d}(x) = b_x$. Similarly, for y and $x+y$ we have

$$\Pr\left[\exists b_y \in \{0,1\} \text{ s.t. } \left|\{i \in [d] : b_y^{(i)} = b_y\}\right| \geq \frac{3d}{4} \,\middle|\, \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \geq 1 - \varepsilon$$

and

$$\Pr\left[\exists b_{x+y} \in \{0,1\} \text{ s.t. } \left|\{i \in [d] : b_{x+y}^{(i)} = b_{x+y}\}\right| \geq \frac{3d}{4} \,\middle|\, \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \geq 1 - \varepsilon\ .$$

Furthermore, if the foregoing events hold, then $\overline{\mathcal{F}}_{t,d}(y) = b_y$ and $\overline{\mathcal{F}}_{t,d}(x+y) = b_{x+y}$.

By applying Lemma 9.11 with $r = d/8$ we have

$$\Pr\left[\left|\{i \in [d] : b_x^{(i)} + b_y^{(i)} = b_{x+y}^{(i)}\}\right| > \frac{3d}{4} \,\middle|\, \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1\right] \geq 1 - \frac{2}{\gamma}\cdot e^{-\frac{(1/8-\rho)^2\cdot\min\{t,d\}}{8}}\ .$$

Note that if all four events hold, then there exists an index $i^* \in [d]$ such that (i) $b_x^{(i^*)} + b_y^{(i^*)} = b_{x+y}^{(i^*)}$ and (ii) $b_x = b_x^{(i^*)}$, $b_y = b_y^{(i^*)}$, and $b_{x+y} = b_{x+y}^{(i^*)}$, which implies that $\overline{\mathcal{F}}_{t,d}(x) + \overline{\mathcal{F}}_{t,d}(y) = \overline{\mathcal{F}}_{t,d}(x+y)$. Therefore, by the union bound we have

$$\Pr_{\overline{\mathcal{F}}_{t,d}}[\overline{\mathcal{F}}_{t,d}(x) + \overline{\mathcal{F}}_{t,d}(y) = \overline{\mathcal{F}}_{t,d}(x+y)] \geq 1 - \frac{6}{\gamma}\cdot e^{-\frac{(1/16-\rho)^2\cdot\min\{t,d\}}{8}} - \frac{3c\cdot d\cdot 2^{-d/16}}{\gamma} - \frac{2}{\gamma}\cdot e^{-\frac{(1/8-\rho)^2\cdot\min\{t,d\}}{8}}\ .$$

This completes the proof of Lemma 9.7. □

9.4 $\overline{\mathcal{F}}_{t,d}$ is close to $\hat{\mathcal{F}}$

In the following claim we prove that $\overline{\mathcal{F}}_{t,d}$ is close to $\hat{\mathcal{F}}$ the the following sense.

Claim 9.12. *Suppose that* $\Pr[\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq 1 - \rho$ *for* $\rho \leq 1/20$. *Let* $S \subseteq \{0,1\}^n$ *be a set of queries. For all events* $E \subseteq \{0,1\}^S$ *it holds that* $\left|\Pr[\hat{\mathcal{F}}(S) \in E] - \Pr[\overline{\mathcal{F}}_{t,d}(S) \in E]\right| \leq (5\,|S| + 2)\rho + \varepsilon$ *for* $\varepsilon = \frac{|S|^2}{t} + O\left(+\frac{|S|}{1-\rho}\cdot(e^{-\Omega(\rho^2\min\{t,d\})} + d\cdot 2^{-5\rho d/4})\right)$.

Proof. We show that for any event $E \subseteq \{0,1\}^S$ it holds that $\Pr[\overline{\mathcal{F}}_{t,d}(S) \in E] \geq \Pr[\hat{\mathcal{F}}(S) \in E] - (5\rho + 3)\,|S| - \varepsilon$. This clearly suffices, since the same bound also holds for the complement event, thus implying the claim.

Consider the sampling procedure of $\overline{\mathcal{F}}_{t,d}$ on the input S. For each $w \in S$ and $i \in [d]$ define $b_w^{(i)} = \mathcal{F}(z_w^{(i)} + w) - \mathcal{F}(z_w^{(i)})$, where $z_w^{(i)}$'s are chosen according to the definition of Enc_d, and let $b_w = \mathrm{maj}\{b_w^{(i)} : i \in [d]\}$.

By Lemma 9.8 for all $w \in S$ we have

$$\Pr[\mathrm{Agr}_{d-r}(w) \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq 1 - \frac{2}{1-\rho} \cdot e^{-\frac{(r/4d-\rho)^2 \cdot \min\{t,d\}}{8}} - \frac{c \cdot d \cdot 2^{-r/4}}{1-\rho} \ .$$

Hence, by letting $r = 5\rho d$ and taking the union bound we get

$$\Pr[\mathrm{Agr}_{(1-5\rho)d}(w)\ \forall w \in S \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq 1 - \varepsilon' \tag{8}$$

for $\varepsilon' = \frac{2|S|}{1-\rho} \cdot e^{-\Omega(\rho^2 \min\{t,d\})} - \frac{c \cdot d \cdot |S| \cdot 2^{-5\rho d/4}}{1-\rho}$. In particular, this implies that

$$\Pr[\mathrm{Agr}_{(1-5\rho)d}(w)\ \forall w \in S] \geq \Pr[\mathrm{Agr}_{(1-5\rho)d}(w)\ \forall w \in S \wedge \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq (1-\varepsilon')(1-\rho) \geq 1-\varepsilon'-\rho \ .$$

Therefore,

$$\begin{aligned}\Pr[\overline{\mathcal{F}}_{t,d}(S) \in E] &= \Pr[(b_w : w \in S) \in E \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \\ &\geq \Pr[(b_w : w \in S) \in E \ \wedge \ \mathrm{Agr}_{(1-5\rho)d}(w)\ \forall w \in S \mid \mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \\ &\geq \Pr[(b_w : w \in S) \in E \ \wedge \ \mathrm{Agr}_{(1-5\rho)d}(w)\ \forall w \in S] - \rho \ .\end{aligned}$$

Observe $\hat{\mathcal{F}}(S)$ can be sampled by choosing distinct $i_w \in [t]$ for each $w \in S$, and outputting $(b_w^{(i_w)} : w \in S)$. Note that if S was a linearly independent set, then we do not need to require i_w's to be distinct. However, since i_w's are all distinct with probability $1 - \frac{|S|^2}{t}$, this is essentially immaterial. Therefore,

$$\begin{aligned}\Pr[\overline{\mathcal{F}}_{t,d}(S) \in E] &\geq \Pr[(b_w : w \in S) \in E \wedge \mathrm{Agr}_{(1-5\rho)d}(w)\ \forall w \in S] - \rho \\ &\geq \Pr[\hat{\mathcal{F}}(S) \in E \wedge \mathrm{Agr}_{(1-5\rho)d}(w)\ \forall w \in S] - 5\rho\,|S| - \frac{|S|^2}{t} - \rho \\ &\geq \Pr[\hat{\mathcal{F}}(S) \in E] - 5\rho\,|S| - \frac{|S|^2}{t} - \varepsilon' - 2\rho \ .\end{aligned}$$

and the claim follows for $\varepsilon = \varepsilon' + \frac{|S|^2}{t}$. □

9.5 Proof of Theorem 7

Let $\mathcal{F}$ be a k-non-signaling function for $k = O(\frac{\bar{k}}{\varepsilon} \cdot (\bar{k} + \log\frac{1}{\varepsilon}))$, and suppose that $\Pr_{x,y,\mathcal{F}}[\mathcal{F}(x) + \mathcal{F}(y) = \mathcal{F}(x+y)] \geq 1 - \varepsilon$ for $\varepsilon \leq 1/400$, and let $\rho = \sqrt{\varepsilon} \leq 1/20$. Then, by Lemma 5.5 we have $\Pr[\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq 1 - \rho$.

By Lemma 9.2 if $k \geq 2d\bar{k} + 3t$, then $\overline{\mathcal{F}}_{t,d}$ is a $\bar{k}$-non-signaling function. By Lemma 9.6 if $\rho \leq 1/20$ and $t = d = O(\bar{k} + \log\frac{1}{\rho} + \log\frac{1}{1-\rho})$ and $\Pr[\mathsf{D}_{t,\rho}(\mathcal{F}(\mathsf{Q}_t)) = 1] \geq 1 - \rho$, then there exists a linear $\bar{k}$-non-signaling function $\mathcal{L}$ such that $\Delta_{\bar{k}}(\mathcal{L}, \overline{\mathcal{F}}_{t,d}) < \rho/2$. By Claim 9.12 if $t = d = O(\frac{1}{\rho^2}(\log|S| + \log\frac{1}{\rho}))$ then for all sets $S \subseteq \{0,1\}^n$ of size at most $\bar{k}$ it holds that

$$\left|\Pr[\overline{\mathcal{F}}_{t,d}(S) \in E] - \Pr[\hat{\mathcal{F}}(S) \in E]\right| \leq (6\,|S| + 2.5)\rho \ .$$

Therefore, if $\varepsilon \leq 1/400$ and $k = O(\frac{\bar{k}}{\varepsilon} \cdot (\bar{k} + \log \frac{1}{\varepsilon}))$, then $\rho = \sqrt{\varepsilon} \leq 1/20$, and we may choose $t = d = O(\frac{1}{\rho^2}(\bar{k} + \log \frac{1}{\rho}))$ so that if $|S| \leq \bar{k}$, then all the foregoing conditions are satisfied, and hence for all events $E \subseteq \{0,1\}^S$ it holds that $\left|\Pr[\hat{\mathcal{F}}(S) \in E] - \Pr[\mathcal{L}(S) \in E]\right| \leq (6\,|S| + 3)\rho$, as required.

A Proof of Lemma 9.10

In the proof we will use Stirling's formula to approximate the binomial coefficients involved.

Proposition A.1 (Stirling's approximation formula)**.** *There exists an absolute constants $c_0 > 0$ such that the following holds.*

1. *For all integers $n > i \geq 1$ it holds that* $c_0 \leq \frac{\binom{n}{i}}{\sqrt{\frac{n}{i\cdot(n-i)}}\cdot\frac{n^n}{i^i\cdot(n-i)^{n-i}}} \leq 1$.

2. *For all integers $n, i, j \geq 1$ such that $n > i + j$ it holds that* $c_0 \leq \frac{\binom{n}{i,j,n-i-j}}{\sqrt{\frac{n}{i\cdot j\cdot(n-i-j)}}\cdot\frac{n^n}{i^i\cdot j^j\cdot(n-i-j)^{n-i-j}}} \leq 1$.

We now prove Lemma 9.10. Let d be an even integer, and let $b_1, \dots, b_d \in \{0,1\}$ be such that not all b_i's are equal, and let $\alpha = \min\{\frac{q}{d}, 1 - \frac{q}{d}\}$ for $q = |\{i \in [d] : b_i = 0\}|$. For an integer $0 < r \leq (1-\alpha)d$ denote by $N_{d,q}(r)$ the number of permutations $\sigma\colon [d] \to [d]$ such that there are exactly r indices $i \in [d/2]$ such that $b_{\sigma(2i-1)} \neq b_{\sigma(2i)}$. Clearly, if $r \not\equiv q \bmod 2$, then $N_{d,q}(r) = 0$. If $r \equiv q \bmod 2$, then

$$N_{d,q}(r) = \binom{d/2}{r,\ \frac{q-r}{2},\ \frac{d-q-r}{2}} \cdot 2^r \cdot q!(d-q)! \ ,$$

where (i) the binomial term reflects the choice of which of the pairs of indices $(2j-1, 2j)$ will have only zeros, only ones, or both, (ii) the term 2^r accounts for the transpositions within the pairs with both a zero and a one, and (iii) $q!(d-q)!$ counts separately the permutations of the zeros and the permutations of the ones in the positions specified by the previous two items.

For an integer $0 \leq r < \alpha d$ denote by $\mathsf{neq}_{d,q}(r)$ the event that there are *exactly* r indices $i \in [d/2]$ such that $b_{\sigma(2i-1)} \neq b_{\sigma(2i)}$, and let $\beta = \frac{r}{d} \in [0, \beta)$. Then

$$\Pr[\mathsf{neq}_{d,q}(r)] = 2^r \cdot \frac{N_{d,q}(r)}{d!} = 2^r \cdot \frac{\binom{d/2}{r,\frac{q-r}{2},\frac{d-q-r}{2}}}{\binom{d}{q}} \tag{9}$$

Next, we show that for all $0 \leq r < \frac{\alpha d}{4}$ it holds that $\Pr[\mathsf{neq}_{d,q}(r)] < c2^{-\alpha d/4}$ for some absolute constant $c > 0$. The claim follows immediately as

$$\begin{aligned}&\Pr_{\sigma}[\text{there are less than } \alpha d/4 \text{ indices } i \in [d/2] \text{ such that } b_{\sigma(2i-1)} \neq b_{\sigma(2i)}]\\ &= \sum_{0 \leq r < \frac{\alpha d}{4}} \Pr[\mathsf{neq}_{d,q}(r)] \leq cd \cdot 2^{-\alpha d/2} \ ,\end{aligned}$$

as required.

The case of $r = 0$: For $r = 0$ and even q the expression in Eq. (9) reduces to

$$\Pr[\mathsf{neq}_{d,q}(r)] = \frac{\binom{d/2}{q/2}}{\binom{d}{q}} \leq \frac{\sqrt{\frac{2d}{q(d-q)}} \cdot \frac{(\frac{d}{2})^{\frac{d}{2}}}{(\frac{q}{2})^{\frac{q}{2}}\cdot(\frac{d-q}{2})^{\frac{d-q}{2}}}}{c_0 \cdot \sqrt{\frac{d}{q(d-q)}} \cdot \frac{d^d}{q^q(d-q)^{d-q}}} = \frac{\sqrt{2}}{c_0} \cdot \frac{q^{q/2}(d-q)^{(d-q)/2}}{d^{d/2}} \ .$$

By substituting $q = (1-\alpha)d$ into the foregoing expression we get

$$\Pr[\mathsf{neq}_{d,q}(r)] \leq \frac{\sqrt{2}}{c_0} \cdot \left[\alpha^\alpha \cdot (1-\alpha)^{1-\alpha}\right]^{d/2} \leq c \cdot 2^{-\alpha d/2} \ ,$$

for the constant $c = \frac{\sqrt{2}}{c_0}$.

The case of $r > 0$: By Stirling's formula, the nominator in the expression in Eq. (9) is upper bounded by

$$\binom{d/2}{r,\, \frac{q-r}{2},\, \frac{d-q-r}{2}} \le \sqrt{\frac{d/2}{\beta d \cdot \frac{\alpha-\beta}{2} d \cdot \frac{1-\alpha-\beta}{2} d}} \cdot \frac{(d/2)^{d/2}}{(\beta d)^{\beta d} \cdot \left(\frac{\alpha-\beta}{2} d\right)^{\frac{\alpha-\beta}{2} d} \cdot \left(\frac{1-\alpha-\beta}{2} d\right)^{\frac{1-\alpha-\beta}{2} d}}$$

$$= \sqrt{\frac{2}{\beta(\alpha-\beta)(1-\alpha-\beta)d^2}} \cdot \left[\frac{1}{(2\beta)^{\beta} \cdot (\alpha-\beta)^{\frac{\alpha-\beta}{2}} \cdot (1-\alpha-\beta)^{\frac{1-\alpha-\beta}{2}}}\right]^d .$$

The denominator is lower bounded by

$$\binom{d}{q} \ge c_0 \cdot \sqrt{\frac{d}{(1-\alpha)d \cdot \alpha d}} \cdot \frac{d^d}{((1-\alpha)d)^{(1-\alpha)d} \cdot (\alpha d)^{\alpha d}}$$

$$= c_0 \cdot \sqrt{\frac{1}{\alpha(1-\alpha)d}} \cdot \left[\frac{1}{(1-\alpha)^{1-\alpha} \cdot \alpha^{\alpha}}\right]^d$$

Therefore, we get the following bound on $\Pr[\mathsf{neq}_{d,q}(r)]$.

$$\Pr[\mathsf{neq}_{d,q}(r)] \le \frac{1}{c_0} \cdot \sqrt{\frac{2\alpha(1-\alpha)}{\beta(\alpha-\beta)(1-\alpha-\beta)d}} \cdot \left[\frac{\alpha^{\alpha} \cdot (1-\alpha)^{1-\alpha}}{(2\beta)^{\beta} \cdot (\alpha-\beta)^{\frac{\alpha-\beta}{2}} \cdot (1-\alpha-\beta)^{\frac{1-\alpha-\beta}{2}}}\right]^d \tag{10}$$

Note that since r is an integer, we have $\beta \in [1/d, \alpha/4]$, and the expression under the square root is upper bounded by

$$\frac{2\alpha(1-\alpha)}{(\beta d)\frac{3\alpha}{4} \cdot (1 - \frac{3\alpha}{4})} = \frac{8}{3} \cdot \frac{1-\alpha}{4-3\alpha} \le \frac{8}{3} .$$

Next, we upper bound the base of the exponent in last term. We first rewrite it is follows.

$$\frac{(1-\alpha)^{1-\alpha} \cdot \alpha^{\alpha}}{(2\beta)^{\beta} \cdot (\alpha-\beta)^{\frac{\alpha-\beta}{2}} \cdot (1-\alpha-\beta)^{\frac{1-\alpha-\beta}{2}}} = \left(\frac{\sqrt{(\alpha-\beta)(1-\alpha-\beta)}}{2\beta}\right)^{\beta} \cdot \left(\frac{\alpha}{\sqrt{\alpha-\beta}}\right)^{\alpha} \cdot \left(\frac{1-\alpha}{\sqrt{1-\alpha-\beta}}\right)^{1-\alpha} .$$

For the first term note that $\sqrt{(\alpha-\beta)(1-\alpha-\beta)} \le \sqrt{\alpha(1-\alpha)} \le 1/2$, and hence

$$\left(\frac{\sqrt{(\alpha-\beta)(1-\alpha-\beta)}}{2\beta}\right)^{\beta} \le \left(\frac{\sqrt{\alpha(1-\alpha)}}{2\beta}\right)^{\beta} \le \left(\frac{1}{4\beta}\right)^{\beta} \le (1/\alpha)^{\alpha/4} ,$$

where the last inequality follows from the fact that the function $f(\beta) = (1/4\beta)^{\beta}$ is monotonically increasing in the interval $[0, 1/4e]$, and hence, for $\alpha < 0.5$ and $\beta \in (0, \alpha/4)$ the maximum is obtained for $\beta_{\max} = \alpha/4$, where $f(\beta_{\max}) = \alpha^{-\alpha/4}$.

For the second term, we have

$$\left(\frac{\alpha}{\sqrt{\alpha-\beta}}\right)^{\alpha} \le \alpha^{\alpha/4} .$$

Indeed, the bound is equivalent to the inequality $\alpha^{3/4} < \sqrt{\alpha - \beta}$, which holds for all $\alpha < 0.5$ and $\beta \leq \alpha/4$.

For the third term, for all $\alpha \leq 0.5$ and $\beta \in [0, \alpha/4]$ we have

$$\left(\frac{1-\alpha}{\sqrt{1-\alpha-\beta}}\right)^{1-\alpha} \leq \left(\frac{1-\alpha}{\sqrt{1-5\alpha/4}}\right)^{1-\alpha} \leq 2^{-\alpha/4} \ .$$

Multiplying the three terms, we get that the last term in Eq. (10) is upper bounded by $2^{-\alpha d/4}$. Therefore, for the constant $c = \frac{8}{3c_0}$ the expression in Eq. (10) satisfies

$$\Pr[\mathsf{neq}_{d,q}(r)] < c2^{-\alpha d/4}$$

for all $r < \alpha d/4$, and the claim follows.

Acknowledgements

We are grateful to Thomas Vidick for suggestions that have improved the presentation in this paper.

References

[ALMSS98] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. "Proof verification and the hardness of approximation problems". In: *Journal of the ACM* 45.3 (1998). Preliminary version in FOCS '92., pp. 501–555.

[AS98] Sanjeev Arora and Shmuel Safra. "Probabilistic checking of proofs: a new characterization of NP". In: *Journal of the ACM* 45.1 (1998). Preliminary version in FOCS '92., pp. 70–122.

[BCHKS96] Mihir Bellare, Don Coppersmith, Johan Håstad, Marcos A. Kiwi, and Madhu Sudan. "Linearity testing in characteristic two". In: *IEEE Transactions on Information Theory* 42.6 (1996), pp. 1781–1795.

[BFL91] László Babai, Lance Fortnow, and Carsten Lund. "Non-Deterministic Exponential Time has Two-Prover Interactive Protocols". In: *Computational Complexity* 1 (1991). Preliminary version appeared in FOCS '90., pp. 3–40.

[BFLS91] László Babai, Lance Fortnow, Leonid A. Levin, and Mario Szegedy. "Checking computations in polylogarithmic time". In: *Proceedings of the 23rd ACM Symposium on Theory of Computing*. STOC '91. 1991, pp. 21–32.

[BGHSV06] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil P. Vadhan. "Robust PCPs of Proximity, Shorter PCPs, and Applications to Coding". In: *SIAM Journal on Computing* 36.4 (2006), pp. 889–974.

[BLR93] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. "Self-Testing/Correcting with Applications to Numerical Problems". In: *Journal of Computer and System Sciences* 47.3 (1993), pp. 549–595.

[CMS18] Alessandro Chiesa, Peter Manohar, and Igor Shinkar. "Testing Linearity against Non-Signaling Strategies". In: *Proceedings of the 33rd Annual Conference on Computational Complexity*. CCC '18. 2018, 17:1–17:37.

[DR04] Irit Dinur and Omer Reingold. "Assignment Testers: Towards a Combinatorial Proof of the PCP Theorem". In: *Proceedings of the 45th IEEE Symposium on Foundations of Computer Science*. FOCS '04. 2004, pp. 155–164.

[FGLSS96] Uriel Feige, Shafi Goldwasser, Laszlo Lovász, Shmuel Safra, and Mario Szegedy. "Interactive proofs and the hardness of approximating cliques". In: *Journal of the ACM* 43.2 (1996). Preliminary version in FOCS '91., pp. 268–292.

[IKM09] Tsuyoshi Ito, Hirotada Kobayashi, and Keiji Matsumoto. "Oracularization and Two-Prover One-Round Interactive Proofs against Nonlocal Strategies". In: *Proceedings of the 24th IEEE Annual Conference on Computational Complexity*. CCC '09. 2009, pp. 217–228.

[Ito10] Tsuyoshi Ito. "Polynomial-Space Approximation of No-Signaling Provers". In: *Proceedings of the 37th International Colloquium on Automata, Languages and Programming*. ICALP '10. 2010, pp. 140–151.

[KRR13] Yael Kalai, Ran Raz, and Ron Rothblum. "Delegation for Bounded Space". In: *Proceedings of the 45th ACM Symposium on the Theory of Computing*. STOC '13. 2013, pp. 565–574.

[KRR14] Yael Tauman Kalai, Ran Raz, and Ron D. Rothblum. "How to delegate computations: the power of no-signaling proofs". In: *Proceedings of the 46th ACM Symposium on Theory of Computing*. STOC '14. Full version available at `https://eccc.weizmann.ac.il/report/2013/183/`. 2014, pp. 485–494.

[KRR16] Yael Tauman Kalai, Ran Raz, and Oded Regev. "On the Space Complexity of Linear Programming with Preprocessing". In: *Proceedings of the 7th Innovations in Theoretical Computer Science Conference*. ITCS '16. 2016, pp. 293–300.

[KT92] Leonid A Khalfin and Boris S Tsirelson. "Quantum/classical correspondence in the light of Bell's inequalities". In: *Foundations of physics* 22.7 (1992), pp. 879–948.

[Kil92] Joe Kilian. "A note on efficient zero-knowledge proofs and arguments". In: *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*. STOC '92. 1992, pp. 723–732.

[McD98] Colin McDiarmid. "Concentration". In: *Probabilistic methods for algorithmic discrete mathematics*. 1998, pp. 195–248.

[Mic00] Silvio Micali. "Computationally Sound Proofs". In: *SIAM Journal on Computing* 30.4 (2000). Preliminary version appeared in FOCS '94., pp. 1253–1298.

[PR17] Omer Paneth and Guy Rothblum. "On Zero-Testable Homomorphic Encryption and Publicly Verifiable Non-Interactive Arguments". In: *Proceedings of the 15th Theory of Cryptography Conference*. TCC '17. 2017.

[PR94] Sandu Popescu and Daniel Rohrlich. "Quantum nonlocality as an axiom". In: *Foundations of Physics* 24.3 (1994), pp. 379–385.

[Ras85] Peter Rastall. "Locality, Bell's theorem, and quantum mechanics". In: *Foundations of Physics* 15.9 (1985), pp. 963–972.

[SA90] Hanif D. Sherali and Warren P. Adams. "A Hierarchy of Relaxations between the Continuous and Convex Hull Representations for Zero-One Programming Problems". In: *SIAM Journal on Discrete Mathematics* 3.3 (1990), pp. 411–430.