On Disperser/Lifting Properties of the Index and Inner-Product Functions[*]

Paul Beame[†]

University of Washington

beame@cs.washington.edu

Sajin Koroth[‡]

University of Victoria

skoroth@uvic.ca

December 2, 2022

Abstract

Query-to-communication lifting theorems, which connect the query complexity of a Boolean function to the communication complexity of an associated 'lifted' function obtained by composing the function with many copies of another function known as a gadget, have been instrumental in resolving many open questions in computational complexity. A number of important complexity questions could be resolved if we could make substantial improvements in the input size required for lifting with the Index function, which is a universal gadget for lifting, from its current near-linear size down to polylogarithmic in the number of inputs N of the original function or, ideally, constant. The near-linear size bound was recently shown by Lovett, Meka, Mertz, Pitassi and Zhang [20] using a recent breakthrough improvement on the Sunflower Lemma to show that a certain graph associated with an Index function of that size is a disperser. They also stated a conjecture about the Index function that is essential for further improvements in the size required for lifting with Index using current techniques. In this paper we prove the following;

- The conjecture of Lovett et al. is false when the size of the Index gadget is less than logarithmic in N.

- The same limitation applies to the Inner-Product function. More precisely, the Inner-Product function, which is known to satisfy the disperser property at size $O(\log N)$, also does not have this property when its size is less than $\log N$.

- Notwithstanding the above, we prove a lifting theorem that applies to Index gadgets of any size at least 4 and yields lower bounds for a restricted class of communication protocols in which one of the players is limited to sending parities of its inputs.

- Using a modification of the same idea with improved lifting parameters we derive a strong lifting theorem from decision tree size to parity decision tree size. We use this, in turn, to derive a general lifting theorem in proof complexity from tree-resolution size to tree-like $Res(\oplus)$ refutation size, which yields many new exponential lower bounds on such proofs.

1 Introduction

In recent years, a substantial number of long-standing problems [8, 9, 19, 11, 23] have been resolved using the method of *lifting*. Lifting results take a gadget function g and show that any function $f : \{0,1\}^N \to \{0,1\}$ that is hard to compute by decision trees can be modified to a new function $F = f \circ g^N$ that is hard for a more powerful computational model, typically that of 2-party communication complexity, in which case $g : X \times Y \to \{0,1\}$ and the inputs for F for the two players are partitioned into $\mathbf{x} \in X^N$ and $\mathbf{y} \in Y^N$.

A particularly natural and important choice of gadget g is the *Index* gadget $\mathrm{IND}_m : [m] \times \{0,1\}^m$ given by $\mathrm{IND}_m(x,y) = y_x$. The Index gadget is universal for all gadgets $g : X \times Y \to \{0,1\}$ where $|X| = |Y| = m$ via the simple reduction where y is replaced by the string $(g(x,y))_{x \in X}$. Since IND_m has a 2-party protocol of cost $\log_2 m + 1$, the communication complexity of $f \circ \mathrm{IND}_m^N$ is at most $O(C^{dt}(f) \log m)$, where $C^{dt}(f)$ is the decision tree complexity of f. Another important gadget g is the Inner-Product function $\mathrm{IP}_b : \{0,1\}^b \times \{0,1\}^b \to \{0,1\}$ given by $\mathrm{IP}_b(x,y) = x \cdot y \bmod 2$.

An important limitation on the quality of lower bounds that can be proven by lifting with a gadget $g : X \times Y \to \{0,1\}$ comes from the fact that the input size for F grows by a factor of $\log_2 |X| + \log_2 |Y|$ bits from that of f. This limits the lower bounds on the lifted function F compared to the input size of F. The original lifting theorems of [21] and [12] used Index gadgets with m a large polynomial in N. Subsequently, [5, 25] proved a lifting theorem for Inner-Product gadgets with $b = c \log_2 N$ for some constant $c > 5$. Later, [5] improved c to almost 2. The first lifting theorem for randomized computation was proved by [13] again for Index but for m an even larger polynomial in N and later again, by [4], for Inner-Product for b a larger constant multiple of $\log_2 N$ than for deterministic lifting.

A key question asked in a precursor paper to these lifting theorems [10] is whether lifting is possible with a sub-logarithmic or even constant-size Inner-Product gadget. Smaller gadgets imply sharper lifting results and more general classes of functions for which lifting may be used to prove lower bounds. Proving such lifting theorems would imply breakthrough results in other areas. For example, proving lifting theorems with constant-size gadgets would give us a near-complete understanding of communication complexity of lifted search problems and would imply breakthrough results in associated areas like proof complexity and circuit complexity[1]. Even improving the gadget size for Index to poly-logarithmic in N would improve the best known monotone circuit size lower bounds [14] from $2^{\tilde{\Omega}(n^{1/3})}$ to $2^{\tilde{\Omega}(n)}$. In the dream range of constant size, by the universality of Index, if there is any lifting theorem for any constant-size gadget, there would be one for constant-size Index gadgets.

Recent work by Lovett, Meka, Mertz, Pitassi and Zhang [20] used a new bound for the Sunflower Lemma [1] to improve the size of the Index gadget that can be used in deterministic lifting results to $O(N \log N)$. They also identified a conjecture regarding entropy deficiency and the disperser property of the Index gadget that is essential for further reductions in gadget size using

---

[1]In this paper we focus on the setting of query-to-communication lifting where the query complexity of f is lifted to the communication complexity of F using the gadget g. There are other lifting theorems (see [24]) which lift analytical parameters of the function f to the communication complexity of F. In these settings, lifting theorems with constant-size gadgets are known [24] but for many interesting applications of lifting, there is a significant gap between analytical parameters of f like approximate-degree (used in [24]) and the query complexity of f. Thus, such lifting theorems with constant-size gadgets are not enough to give the results alluded to above.

current techniques.

Before stating the conjecture of Lovett, Meka, Mertz, Pitassi and Zhang [20] we give an outline of the meta-technique for proving query to communication lifting theorems, known as the simulation theorem framework.

The lifting paradigm

The general paradigm for proving a query-to-communication lifting theorem is a step-by-step simulation argument that begins with a communication protocol Π for $f \circ g^N$ on inputs in $X^N \times Y^N$ and derives a decision tree T computing f on inputs $z \in \{0, 1\}^N$.

Beginning at the root of Π and with T a single root node, the simulation proceeds to follow a path in Π maintaining sets of inputs $\mathcal{X} \subseteq X^N$ and $\mathcal{Y} \subseteq Y^N$ consistent with the current node u in protocol Π. (The exact procedure for choosing the path and the sets $\mathcal{X}$ and $\mathcal{Y}$ varies.)

At any point in time when $\mathcal{X} \times \mathcal{Y}$ has revealed too much about the value of $z_i = g(x_i, y_i)$ for some i, the simulation at the current leaf node v of T queries z_i and adds the children v' and v'' to T, one for each outcome. The simulation then splits into cases depending on whether the 0 or 1 out-edge from v is being followed. There may be multiple i for which this may need to be done at the same time.

The simulations maintain several invariants at each corresponding pair of nodes u in Π and v in T that occur in this simulation. In particular, if $\mathcal{X}$ and $\mathcal{Y}$ are associated with this pair of nodes and $I \subseteq [N]$ is the set of input indices queried on the path in T to v and z_I is the assignment that takes T to the node v then we require

$$g^I(\mathcal{X}, \mathcal{Y}) = z_I \qquad \text{(Consistency)}$$
$$g^{[N] \setminus I}(\mathcal{X}, \mathcal{Y}) = \{0, 1\}^{[N] \setminus I}. \qquad \text{(Disperser/Extensibility)}$$

The consistency property is obviously required for correctness. The disperser property is required because the simulation cannot predict what query indices will be needed for T in the future. Overall, in order to yield a good complexity bound, the argument also has to bound the length of the path to v in T, which is the size of the set I, as a function of the length of the path from the root to u in Π.

In order to maintain these properties, the simulations also maintain some "nice" structure on the sets $\mathcal{X}$ and $\mathcal{Y}$. The most common notions of nice structure are small *entropy deficiency* of the induced distributions on the unqueried coordinates $[N] \setminus I$ or high *min-entropy rate* (equivalently the *block min-entropy*)[2]. The min-entropy rate is the minimum ratio of the min-entropy of the induced distributions on any subset of unqueried blocks compared to the maximum possible entropy on those blocks.

There are other "nice" properties that were used in the past. For example, one of the first lifting theorems by Raz and McKenzie [21] used a combinatorial notion of niceness defined as average-degrees in a layered graph corresponding to $\mathcal{X}$. This property was also used in the later reproving of the result by [12], and the result on extending deterministic lifting theorems to a larger class of

---

[2]Many existing results for the Index gadget use bounds on the min-entropy rate on the X^N side and entropy deficiency on the Y^N side.

gadgets including Inner Product by [25, 5][3]. All of the results using this combinatorial property crucially depend on a transformation in layered graphs from average degree to minimum degree known as the "thickness lemma" to prove the disperser property of the gadget. It is a folklore result that such average-degree to min-degree transformations do not work for Index gadgets of linear size. Thus, using "average-degree" as the nice property cannot yield lifting theorems with sublinear-size Index gadgets using existing techniques.

The LMMPZ conjecture on entropy deficiency and disperser properties of IND

The conjecture of Lovett et al. [20] is a necessary condition for small entropy deficiency to be sufficient for lifting with the Index function. To motivate the parameters of the conjecture we first note how entropy deficiency relates to the numbers of bits of communication sent in the protocol Π:

In the course of following a path in Π to a node u, each bit communicated may split the set of consistent inputs in either X^N or Y^N by a factor of 2, which increases the entropy deficiency by 1. If good min-entropy rate is also required, additional pruning must be done, which further increases the entropy deficiency. Therefore, the best one can do in terms of maintaining small entropy deficiency is to maintain a bound Δ on entropy deficiency for $\mathcal{X}$ and $\mathcal{Y}$ that is proportional to the number of bits sent in Π. Bounding the length of the path in the decision in terms of the number of bits sent means that $|I|$ should not be too large as a function of Δ. This led Lovett et al. to formulate the following conjecture on the disperser properties of Index as a first step towards obtaining lifting theorems for small gadget sizes:

Conjecture 1.1 ([20, Conjecture 11]). *There exists c, such that for all large enough m the following holds: Let $\mathcal{X}, \mathcal{Y}$ be distributions on $[m]^N$, $(\{0,1\}^m)^N$, respectively, each with entropy deficiency at most Δ. Then $IND_m^N(\mathcal{X}, \mathcal{Y})$ contains a sub-cube of co-dimension at most $c\Delta$. That is, there exists $I \subseteq [N]$, $|I| \leq c\Delta$, and $\gamma \in \{0,1\}^I$ such that for all $z \in \{0,1\}^N$ with $z_I = \gamma$ we have*

$$\Pr_{x \sim \mathcal{X}, y \sim \mathcal{Y}}[IND_m^N(x, y) = z] > 0. \tag{1}$$

Remark: Note that the condition Equation (1) is somewhat weaker than the combination of the consistency and disperser conditions in the above lifting paradigm. The lifting paradigm would correspond to additionally requiring that the (x, y) pair in Equation (1) come from some $\mathcal{X}' \subseteq \mathcal{X}$ and $\mathcal{Y}' \subseteq \mathcal{Y}$ such that $\text{IND}_m^I(\mathcal{X}', \mathcal{Y}') = \gamma$. Here, one could satisfy Equation (1) using pairs (x, y) and (x', y') with $\text{IND}_m^I(x, y) = \text{IND}_m^I(x', y') = \gamma$ but $\text{IND}_m^I(x, y') \neq \gamma$.

The results in [20] prove the conjecture for $m = O(N \log N)$, in fact the stronger version with separate consistency and disperser properties required for lifting; previously it was only known when $m \gg N^2$. Based on a related statement about p-biased $(\mathcal{X}, \mathcal{Y})$ proved in the Robust Sunflower Theorem from [1], the authors [20] also suggest that it is hopeful to prove the conjecture when $m = \text{poly}(\log N)$ using techniques from their work and [1].

---

[3][5] uses it slightly differently from the application of the property for the Index gadget.

Our results

We disprove the LMMPZ conjecture when m is $\log_2 N - \omega(1)$, even when $\mathcal{X}$ and $\mathcal{Y}$ are also assumed to have extremely high min-entropy rate. In our counterexample the distribution for $\mathcal{X}$ is uniform on $[m]^N$ and so has full entropy and maximum possible min-entropy rate. The distribution on $\mathcal{Y}$ is also uniform so we view both $\mathcal{X}$ and $\mathcal{Y}$ as subsets of $[m]^N$ and $(\{0,1\}^m)^N$ respectively.

Though the parameter Δ governing the entropy deficiency in the conjecture is universally quantified, the failure of the conjecture occurs over a very wide range of values of Δ. In fact, when $m \leq (1-\alpha) \log_2 N$, we prove a much larger gap and show that $|I|$ must be $\Omega(N^\alpha)$ independent of Δ for Equation (1) to hold.

Theorem 1.2. *For any $\Delta \geq 1$ and m with $2^m \leq N/(K\Delta)$ for $K \geq 1$ there is a set $\mathcal{Y} \subseteq (\{0,1\}^m)^N$ of entropy deficiency at most Δ and min-entropy rate at least $1 - 1/m$ such that for every $I \subseteq [N]$ with $|I| \leq (K-1)\Delta$, the set $\text{IND}_m^{N \setminus I}([m]^N, \mathcal{Y})$ does not contain the all-0 string.*

Since $\mathcal{X} = [m]^N$ has no deficiency (and min-entropy rate 1) we immediately derive the following:

Corollary 1.3. *Conjecture 1.1 is false when $m \leq \log_2(N/\Delta) - \omega(1)$. Moreover, for all Δ, when $m \leq (1-\alpha)\log_2 N$, for any set I, $|I|$ must be $\Omega(N^\alpha)$ for Equation (1) to hold.*

Furthermore, an analogous property applies to the Inner-Product function:

Theorem 1.4. *For $b \leq \log_2(N/(K\Delta))$ for $K \geq 1$, there is a set $\mathcal{Y} \subset (\{0,1\}^b)^N$ of entropy deficiency at most Δ and min-entropy rate more than $1 - 1/b$ for every $I \subseteq [N]$ with $|I| \leq (K-1)\Delta$, the set $\text{IP}_b^{N \setminus I}((\{0,1\}^b)^N, \mathcal{Y})$ does not contain the all-1 string.*

Therefore, though lifting theorems, both deterministic and randomized, have already been proven for Inner-Product gadgets on $c \log_2 n$ bits using only properties of small entropy-deficiency and high min-entropy rate [4], using these properties we can at best reduce the Inner-Product gadget size in such lifting theorems by at most a constant factor since lifting for Inner-Product gadgets with significantly fewer than $\log_2 n$ bits are impossible using those properties.

The proof idea for these theorems is quite simple and relies on the fact that for such small values of m, it is likely that a uniformly random string y will have many blocks i where $y_i = 0^m$ and hence cannot have 1 output values in any of those coordinates.

Despite this setback, the dream of lifting theorems for constant-size gadgets remains. Our second main result is that, though we can rule out the disperser properties of Conjecture 1.1 for IND_m with sub-logarithmic m, there is an interesting class of protocols, one in which Bob's messages are constrained to be parity functions of his input string y, in which we can prove a deterministic lifting theorem using IND_m gadget for *constant size m*.

Since Alice is unrestricted and Bob is restricted, we call such protocols semi-structured protocols. We obtain a lifting theorem for semi-structured protocols showing that the decision tree height is asymptotically at most a $1/\log m$ fraction of the complexity of the communication protocol for the lifted function with IND_m^N. As is typical for deterministic lifting theorems, this works both for functions and for search problems. For protocols in which both Alice and Bob only send

parities of their inputs, we obtain an even stronger simulation that applies to the size of the decision tree produced in terms of the number of leaves (size) of the communication protocol for the lifted function.

In particular a modification of this idea gives us a generic theorem that lifts decision tree lower bounds of height t or size s for any explicit function f on n inputs to a corresponding lower bound for parity decision trees of height $\Omega(t)$ or size $\Omega(s)$ for an explicit function f' on $O(n)$ inputs.

The latter also yields new lower bounds for tree-like proofs in the $Res(\oplus)$ proof system (also known as $ResLin_2$) introduced by Raz and Tzamaret [22]. Itsykson and Sokolov [16, 17] had previously shown tight exponential lower bounds for the pigeonhole principle as well as exponential lower bounds for a very restricted kind of lifted formula based on Tseitin formulas from [2]. Huynh and Nordström [15] gave lifting theorems for a variety of other proof systems using constant-size Index gadgets (indeed with $m = 3$) but these only yield good bounds for a restricted class of formulas whose search problems have high "critical block sensitivity". Here we obtain exponential lower bounds for a substantially broader class of formulas. In particular, for any of the vast class of k-CNF formulas φ for which exponential tree-resolution lower bounds are known, we obtain lifted $O(k)$-CNF formulas φ' with a constant factor increase in number of variables (and a constant factor increase in number of clauses if k is constant) requiring tree-like $Res(\oplus)$ refutations of exponential size (indeed at least the tree-like resolution refutation size for φ).

Related Work   Independently of our work, Chattopadhyay, Mande, Sanyal, and Sherif [6] have obtained closely related lifting results for parity decision tree size and the size of tree-like $Res(\oplus)$ proofs. Their lifting theorem works not only for lifting with Index gadgets but, more generally, for lifting with a class of gadgets that includes Inner-Product and other simple gadgets. Their methods and ours have considerable similarity, particularly in the use of row-reduction as a key component.

2   Preliminaries

Notation: For a set of vectors (or a distribution $\mathcal{X}$) on U^N and $I \subseteq [N]$, we use $\mathcal{X}_I$ to denote the projection of $\mathcal{X}$ onto the coordinates in I. For a function h on U, we let $h^I(\mathcal{X})$ denote the set of all possible vectors of outputs of h^I on $\mathcal{X}_I$; if this set is a singleton w we abuse notation and simply define the value to be w.

Information theory: We use several definitions for forms of entropy:

Definition 2.1. *The* entropy deficiency *(sometimes simply* deficiency*) of a distribution $\mathcal{X}$ on a universe U, $D_\infty(\mathcal{X})$, is $\log_2 |U| - H_2(\mathcal{X})$. For a subset $\mathcal{V} \subseteq U$, the deficiency of $\mathcal{V}$ is that of the uniform distribution on $\mathcal{V}$. In particular, for example, the deficiency of a set of inputs $\mathcal{Y} \subseteq (\{0,1\}^m)^N$ of Bob satisfies*

$$2^{-D_\infty(\mathcal{Y})} = \frac{|\mathcal{Y}|}{2^{m \cdot N}}.$$

Definition 2.2. *The* min-entropy *of a distribution $\mathcal{X}$ on U, $H_\infty(\mathcal{X})$, is*

$$\min_{x \in U} \log_2(1/\Pr_{\mathcal{X}}(x)).$$

For a distribution $\mathcal{X}$ on a set U^N, the min-entropy rate *of $\mathcal{X}$ is the maximum τ such that for every $J \subseteq [N]$, $H_\infty(\mathcal{X}_J) \geq \tau|J|\log_2|U|$ or, equivalently, such that for all $\alpha_J \in U^J$,*

$$\Pr_{x \sim \mathcal{X}}[x_J = \alpha_J] \leq |U|^{-\tau \cdot |J|}.$$

We use the following fundamental fact about the effect of conditioning on the min-entropy of a random variable.

Fact 2.3. *Let $\mathcal{X}$ be a random variable and let E be an event. Then $H_\infty(\mathcal{X} \mid E) \geq H_\infty(\mathcal{X}) - \log_2(1/\Pr[E])$.*

Probability: We will use the following nice bound on the median of any binomial distribution.

Proposition 2.4 ([18]). *The median of a binomial distribution $B(n, p)$ lies between $\lfloor np \rfloor$ and $\lceil np \rceil$.*

Linear algebra: We use row-reduced form to represent matrices in our simulation theorems.

Definition 2.5 (Row-reduced matrices). *A matrix M with r rows is said to be* row reduced *if it contains an $r \times r$ identity submatrix.*

Parity decision trees and $Res(\oplus)$ refutations: A *parity decision tree* over a set of Boolean variables Z defining a space $\{0, 1\}^Z$ of Boolean vectors is a rooted binary tree in which each internal node is labeled by a parity of variables from Z with out-edges labeled 0 and 1 respectively. Each leaf is labeled by an output value. Such a decision tree computes a function on $\{0, 1\}^Z$.

For a function or relation f on Boolean inputs, let $C^{dt}(f)$ and $\text{size}^{dt}(f)$ be the minimal height and size, respectively of any decision tree computing f and let $C^{\oplus dt}(f)$ and $\text{size}^{\oplus(dt)}(f)$ be the corresponding measures for parity decision trees.

Before defining the proof system $Res(\oplus)$ and its relationship to parity decision trees, we first review the resolution proof system and its relationship to ordinary decision trees.

A *resolution* (*Res*) refutation of an unsatisfiable CNF formula φ on variables Z is a sequence of clauses ending in the empty clause $\bot$ in which each clause is either a clause of φ, or follows from two prior clauses using the inference rule

$$\frac{A \vee z_i, \ B \vee \overline{z_i}}{A \vee B}$$

for some variable $z_i \in Z$; we say that this step *resolves* on variable z_i.

This sequence yields a directed acyclic graph (dag) of in-degree 2 each of whose nodes is labeled by a clause on the variables in Z, with sources labeled by clauses of φ and sink labeled by $\bot$. The resolution refutation is *tree-like* if this associated dag is a tree and the associated refutation is called a *tree-resolution* refutation of φ. For an unsatisfiable CNF formula φ, let $\text{tree}_{Res}(\varphi)$ be the minimum size of any tree-resolution refutation of φ. Further, let $\text{width}_{Res}(\varphi)$ denote the minimum over all resolution refutations of φ of the length of the longest clause in the refutation. We have the following result of Ben-Sasson and Wigderson [3].

6

Proposition 2.6. *For any CNF formula φ with clause-size at most k, $\text{tree}_{Res}(\varphi) \geq 2^{\text{width}_{Res}(\varphi)-k}$.*

Every unsatisfiable CNF formula φ yields an associated total search problem Search_φ which takes as input an assignment $z \in \{0,1\}^Z$ and produces the name of a clause that is falsified by z.

Proposition 2.7. *Given any unsatisfiable formula φ, minimal tree-resolution refutations of φ and decision trees solving Search_φ are isomorphic. This isomorphism identifies nodes that resolve on a variable z_i with those that branch on variable z_i.*

Each node in a decision tree naturally corresponds to the sub-cube of the input set $\{0,1\}^Z$ given by the constraints on the path to the node from the root. Similarly, any clause C can be identified with a sub-cube of $\{0,1\}^Z$ consisting of the set of all inputs falsified by C. We say that a clause C is a *weakening* of a clause A iff there is some clause B such that $C = A \vee B$; alternatively this is equivalent to saying that the sub-cube corresponding to C is contained in the sub-cube corresponding A. With this correspondence, the isomorphism in the above proposition means that the sub-cube for each node in the decision tree corresponds to a weakening of the clause at the isomorphic node in the tree-resolution refutation.

A *Res($\oplus$) refutation* of an unsatisfiable CNF formula φ in variables Z is a sequence of affine subspaces of $\mathbb{F}_2^Z$ ending in the subspace $\mathbb{F}_2^Z$ such that each subspace in the list is either the sub-cube corresponding to a clause of φ or follows from two prior subspaces A and B via the inference rule:

$$\frac{A,\ B}{C} \qquad \text{if } C \subseteq A \cup B.$$

(Alternatively, one can replace each subspace by an expression for its dual, namely a disjunction of parity equations over $\mathbb{F}_2$, each of which is the negation of one of a set of linear equations defining the affine subspace. Therefore, each line is a clause over parities and the subspace is the set of inputs that falsifies it. In this way, the single inference rule generalizes the weakening rule of resolution and generalizes resolving on a single literal to resolving on a parity of variables; it is not hard to see that the subspace C must be contained entirely in A or in B unless there is a unique linear constraint defining A whose negation is a defining equation for B. This representation is not unique since there may be many different choices of defining equations for an affine subspace, but we assume that the proof is representation independent. As noted by Isykson and Sokolov [16, 17], the semantic view of $Res(\oplus)$ refutations we have presented is a standard proof system in the sense of Cook and Reckhow [7] since inference is explicitly verifiable in polynomial time.)

As with resolution refutations, we can define the dag of indegree (at most) 2 associated with a $Res(\oplus)$ refutation and let *tree-Res($\oplus$)* be the proof system consisting of $Res(\oplus)$ refutations whose associated dag is a tree. For an unsatisfiable CNF formula φ, let $\text{tree}_{Res(\oplus)}(\varphi)$ be the minimum size of any *tree-Res($\oplus$)* refutation of φ. Using the analogous ideas to the isomorphism between decision trees and minimal tree-resolution proofs, Itsykson and Sokolov proved the following correspondence:

Proposition 2.8. *[16, 17] Given any unsatisfiable formula φ, minimal tree-Res($\oplus$) refutations of φ and parity decision trees solving Search_φ are isomorphic. This isomorphism identifies nodes that resolve on a parity function $\oplus_S(z)$ with those that branch on $\oplus_S(z)$.*

Lifting CNF formulas with IND$_m$: There are a few options for how to do this. Since we will use this when $m = 2^\ell$ is a constant, we choose a simple option that has $N\ell$ Boolean variables $x_{i,j'}$ for $j' \in \{0, \ldots, \ell - 1\}$ and $i \in [N]$ and Nm Boolean variables $y_{i,j}$ for $j \in \{0, \ldots, m - 1\}$ and $i \in [N]$. As usual, the interpretation we have for IND$_m^N$ is that $z_i = y_{i, x_{i,\ell-1} \ldots x_{i,0}}$.

Given a k-CNF formula φ in N variables $Z = \{z_1, \ldots, z_N\}$, we define an $(\ell + 1)k$-CNF formula $\varphi \circ \text{IND}_m^N$ on the $x_{i,j}$ and $y_{i,k}$ variables as follows: Each clause $(z_{i_1}^{b_1} \vee \cdots z_{i_k}^{b_k})$ of φ is replaced by m^k clauses of length $(\ell + 1)k$, one for each tuple $(j^{(1)}, \ldots, j^{(k)})$, which expresses the statement that if the ℓ bits $x_{i_1,*}$ encode value j_1, those of $x_{i_2,*}$ encode j_2, ... and those of $x_{i_k,*}$ encode j_k, then $y_{i_1,j_1}^{b_1} \vee \cdots y_{i_k,j_k}^{b_k}$ must be true. With this definition, each clause of $\varphi \circ \text{IND}_m^N$ corresponds to a unique clause of φ. Moreover, a falsified clause of $\varphi \circ \text{IND}_m^N$ on one of its input vectors yields a falsifying assignment to the corresponding clause of φ under the vector of z values given by IND$_m^N$.

3 On the insufficiency of low deficiency and high min-entropy rate

In this section we prove Theorems 1.2 and 1.4.

Let K be any function of N with $K \geq 1$ for all N and assume that $2^m \leq N/(K\Delta)$. We will construct a specific distribution $\mathcal{Y}$, with deficiency bounded by Δ and min-entropy rate at least $1 - 1/m$, such that for any $I \subseteq [N]$ with $|I| \leq (K-1)\Delta$, $\text{IND}_m^{[N] \setminus I}([m]^N, \mathcal{Y})$ does not contain the all-1 string.

We establish this simply by showing that for all $(x, y) \in [m]^N \times \mathcal{Y}$, $\text{IND}_m^N(x, y)$ has Hamming weight more than $(K-1)\Delta$. Thus any projection of $\text{IND}(\mathcal{X}, \mathcal{Y})$ onto $N - |I| \geq N - (K-1)\Delta$ coordinates will contain at least one 1, and would therefore miss the all-0 string.

To this end, we will construct $\mathcal{Y}$ as the uniform distribution on a subset $S \subseteq (\{0,1\}^m)^N$ with the following properties :

1. Every $y \in S$ has at least $k > (K-1)\Delta$ blocks that are equal to 1^m.

2. $|S| \geq 2^{mN - \Delta}$

3. The min-entropy on any subset of $b \leq N$ blocks is at least $(m-1)b$.

The second property ensures that the deficiency of $\mathcal{Y}$ is at most Δ and the first property guarantees that every output has at least k 1-bits no matter what the input x is. The third property is simply that the min-entropy rate is at least $(1 - 1/m)$.

The essence of the proof idea applies in the case that $\Delta = 1$; the general case is a simple extension of that special case:

Counterexample when $\Delta = 1$: We derive a counterexample in this case by choosing $k = \lfloor K \rfloor > K - 1$ and setting S to be the set of all inputs in $(\{0,1\}^m)^N$ that have at least k blocks of the form 1^m; i.e., all-1 blocks.

The key observation that makes this work is the following:

Observation 3.1. *For y chosen uniformly at random from $(\{0,1\}^m)^N$, the number of all-1 blocks in y is distributed according to the binomial distribution $B(N, 1/2^m)$.*

In particular, since $\Delta = 1$ this means that the expected number of all-1 blocks in y chosen uniformly from $(\{0,1\}^m)^N$ is $N/2^m \geq K$. By applying a bound on the median of binomial distributions, we obtain the following:

Lemma 3.2. *For $2^m \leq N/K$, at least $1/2$ of all strings in $(\{0,1\}^m)^N$ have more than $K - 1$ all-1 blocks.*

Proof. By Observation 3.1, the number of all-1 blocks is given by the binomial distribution $B(N, 1/2^m)$. By Proposition 2.4, the median of this distribution is at least $\lfloor N/2^m \rfloor \geq \lfloor K \rfloor > K - 1$. The claim follows since the binomial is integer-valued. $\square$

Lemma 3.2 shows that $\mathcal{Y}$ is a uniform distribution with deficiency at most 1, which means that the projection on any b blocks has min-entropy at least $mb - 1$ and hence $\mathcal{Y}$ has min-entropy rate at least $1 - 1/m$.

Counterexample for $\Delta > 1$ but $o(N)$: For this we assume without loss of generality that Δ is an integer and $N = N'\Delta$ for some integer N' since rounding can add only $\Delta - 1$ extra coordinates.

Since $2^m \leq N/(K\Delta)$, we have $2^m \leq N'/K$. This means that we can use the counterexample distribution $\mathcal{Y}'$ for the case $\Delta = 1$ on N' coordinates. We define $\mathcal{Y}$ to the direct product of Δ independent copies of the distribution $\mathcal{Y}'$ on disjoint coordinates.

By construction, $\mathcal{Y}$ has deficiency at most Δ; it also has min-entropy rate at least that of each $\mathcal{Y}'$ which is $1 - 1/m$ since it is a product over disjoint coordinates. Also by construction, every y in the support of $\mathcal{Y}$ has more than $(K - 1)\Delta$ all-1 blocks which means that no set I of at most $(K - 1)\Delta$ coordinates cannot cover all of the all-1 blocks of y, which would be necessary to have the all-0 string in $\text{IND}_m^{[N]\setminus|I|}([m]^N, \mathcal{Y})$.

Extending the counterexamples to other gadgets We note that the above result also disproves a similar conjecture for any 2-party gadget $g : X \times Y \to \{0,1\}$ whose communication matrix has a row or column that has a constant value. For example, the Inner-Product gadget on $\text{IP}_b : \{0,1\}^b \times \{0,1\}^b$ has this property for the row or column of its communication matrix indexed by 0^b. That is, $\langle x, 0^b \rangle = 0$ for all $x \in \{0,1\}^b$. It is easy to see that the above analysis works to disprove the analogous conjecture for Inner-Product under the same conditions, though in this case, the output vector that would be missed is the all-1 vector.

4 Lifting theorem for semi-structured protocols

Since Conjecture 1.1 is false for $m = (1 - o(1)) \log N$, without modification, existing techniques cannot reduce the gadget size below this threshold. However, this does not rule out other approaches to proving lifting theorems with very small gadgets, even for constant-size ones. Are such theorems with constant-sized gadgets possible at all?

As a first step towards answering this question, in this section we prove a (deterministic) lifting theorem with constant-size Index gadgets for a restricted family of communication protocols.

The restricted family that we consider are deterministic communication protocols in which Alice is unrestricted, but Bob is only allowed to communicate parities of his input bits. Since Alice is unrestricted and Bob is restricted, we call such protocols *semi-structured* or $(*, \oplus)$-protocols.

For a Boolean function $F : X \times Y \to \{0, 1\}$, we use $C^{*,\oplus}(F)$ to denote the deterministic communication complexity of F by such protocols. Similarly, for a relation (search problem) $R \subseteq X \times Y \times W$ we use $C^{*,\oplus}(R)$ to denote the deterministic communication complexity of such protocols solving R, that is, when Alice receives $x \in X$ and Bob receives $y \in Z$, the protocol outputs some $w \in W$ with $(x, y, w) \in R$ or outputs $\perp$ if no such w exists.

Theorem 4.1 (Lifting theorem for semi-structured protocols). *Let $m \geq 4$ be an integer. For every* $f : \{0, 1\}^N \to \{0, 1\}$,

$$C^{*,\oplus}(f \circ IND_m^N) \geq \frac{1}{2} C^{dt}(f) \log_2 m.$$

Furthermore for every $R \subseteq Z^N \times W$,

$$C^{*,\oplus}(R \circ IND_m^N) \geq \frac{1}{2} C^{dt}(R) \log_2 m.$$

Let Π be a $(*, \oplus)$-protocol for $R \circ IND_m^N$ of complexity $C^{*,\oplus}(R)$. Without loss of generality we can assume that $C^{*,\oplus}(R) \leq N(\log_2 m + 1)$. Following the lifting theorem paradigm we prove Theorem 4.1 by showing how to produce a decision tree T for R of height at most that of Π by simulating Π.

4.1 High level overview and invariants

Since Bob is only allowed to communicate parity equations, we will follow the lifting paradigm and maintain the set $\mathcal{Y}$ as an affine subspace over $\mathbb{F}_2^{[N] \times [m]}$. We will also maintain the property that the codimension d of $\mathcal{Y}$ is at most the total number of bits communicated during the protocol.

At every point in our simulation we will maintain a set $D_{\mathcal{Y}} \subseteq [N] \times [m]$ of *dependent* coordinates. All other coordinates in $[N] \times [m]$ will be *free*. We maintain $\mathcal{Y}$ of codimension d as the set of a solutions of a system $\mathcal{E}$ of d affine equations over $\mathbb{F}_2$ in *row-reduced* form

$$\mathbf{M} \cdot y = \mathbf{b},$$

such that $\mathbf{M}$ (up to permutation of rows) is a $d \times d$ identity submatrix on the columns $D_{\mathcal{Y}}$ that we have designated as dependent coordinates. It is immediate from this set-up that we have the properties:

(A) $|D_{\mathcal{Y}}| = \text{codim}(\mathcal{Y})$.

(B) For every total assignment to the free coordinates, there is a unique assignment to $D_{\mathcal{Y}}$ that extends it to an element of $\mathcal{Y}$.

Bob's communication of some parity function $\oplus_{(i,j) \in S} y_{i,j}$ of his input can add at most one new affine equation to $\mathcal{E}$, depending on whether or not the value of that parity function is already fixed on $\mathcal{Y}$. There is no change to $\mathcal{E}$ if and only if the parity is in $\text{span}(\mathcal{E})$, the $\mathbb{F}_2$ span of the parities defining $\mathcal{E}$.

As in the general lifting paradigm, we maintain a set $I \subseteq [N]$ of *fixed* indices[4] on which there is a single fixed output for $IND_m^I(\mathcal{X}, \mathcal{Y})$. In addition to the above, we also maintain the following invariants.

---

[4] As in the discussion so far, to keep notions separate we will use the term "indices" to refer to elements of $[N]$ and "coordinates" to refer to elements of $[N] \times [m]$.

(C) For all indices that are not fixed, elements of $\mathcal{X}$ only point to free coordinates; that is, for every $i \in [N] \setminus I$, and every $x \in \mathcal{X}$ we have $(i, x_i) \notin D_{\mathcal{Y}}$.

(D) $\mathcal{X}$ has min-entropy rate at least $\tau = 1/2$ on $[N] \setminus I$.

Maintaining (C) is quite easy: Whenever we identify a new dependent coordinate $(i, j) \in [N] \times [m]$ in $\mathcal{Y}$, we simply update $\mathcal{X}$ by removing all $x \in \mathcal{X}$ with $x_i = j$.

We maintain (D) using the methods of Göös, Pitassi, and Watson (GPW) to restore the min-entropy rate whenever it falls too low. Since the affine structure of $\mathcal{Y}$ allows for a precise definition of dependent coordinates, our lifting theorem differs from the GPW-style lifting theorems in its parameters and philosophy[5]. These differences allow us to overcome the dependence between m and N in such theorems.

4.2 The Simulation Algorithm

As we discussed in our high-level overview, we maintain $\mathcal{Y}$ as an affine subspace of $(\{0,1\}^m)^N$ defined by a set of linearly independent equations $\mathcal{E}$ in a row-reduced form that contain an identity matrix on the set $D_{\mathcal{Y}}$ of dependent coordinates. We need to be able to update this as new affine equations are added, either because of communication by Bob or because we have added some i to the set I of fixed indices.

For this, we define a helper function **row-reduce**$(\mathcal{E}, D_{\mathcal{Y}}, e)$ that takes as input

- a set of row-reduced equations $\mathcal{E}$,

- a set $D_{\mathcal{Y}}$ of coordinates for its dependent variables, and

- a new affine equation e linearly independent of $\mathcal{E}$,

and uses Gaussian elimination to return a pair $(\mathcal{E}', (i, j))$ where $\mathcal{E}'$ is equivalent to $\mathcal{E} \cup \{e\}$, $(i, j) \notin D_{\mathcal{Y}}$ and $\mathcal{E}'$ is row-reduced, as witnessed by the columns of $D_{\mathcal{Y}} \cup \{(i, j)\}$. We note that when a new equation is introduced while adding i to the set of fixed indices, the new equation e will be of a particularly simple form, namely $y_{i,j} = b$ where $(i, j) \notin D_{\mathcal{Y}}$ by our maintenance of invariant (C), in which case the new dependent coordinate returned will be (i, j).

We follow a variant of the simulation algorithm of GPW[12], with a few modifications to identify and exclude dependent coordinates in $\mathcal{Y}$. The algorithm uses a sub-routine **RestoreMinEntropy-RateAndQuery** (see Algorithm 2). This is essentially density restoration from GPW and makes sure that $\mathcal{X}$ has min-entropy rate at least $\tau = 1/2$ by adding fixed indices to I, adding queries to the decision tree and fixing some coordinate y_{i,α_i} to the query answer for z_u. Note that GPW style lifting only uses this procedure to restore min-entropy rate at a node where Alice speaks, but we also may need this when Bob speaks because we reduce $\mathcal{X}$ by removing pointers to dependent coordinates. Another difference in our version of density-restoration is that we only chose the first part in the partition (as we are doing deterministic lifting opposed to the randomized lifting in [13]).

---

[5] GPW never makes queries based on Bob's communication, but we do!

1 Algorithm: Query$_\Pi(z)$

  Data: $z \in \{0,1\}^N$, $\mathcal{X} = [m]^N$, $\mathcal{Y} = (\{0,1\})^N$, $\mathcal{E} = \varnothing$, $D_\mathcal{Y} = \varnothing$, $I = \varnothing$, $\rho = *^n$, protocol Π for
    $R \circ \mathrm{IND}_m^N$, v=root of Π.

  Result: element of $R(z)$.

2 while *v is not a leaf* **do**

3   Let v_0, v_1 be the children of v following communication 0 and 1 respectively

4   **if** *Bob speaks at v* **then**

5   Let the parity function at v be $\oplus_{(i,j) \in S} y_{i,j}$

6   **if** $\oplus_{(i,j) \in S} y_{i,j} \in span(\mathcal{E})$ **then**

7 $v \leftarrow v_b$ for the unique b such that $\oplus_{(i,j) \in S} y_{i,j} = b$ for all $y \in \mathcal{Y}$

8   **else**

          `// Half of` $\mathcal{Y}$ `goes to` v_0`, half to` v_1`; we choose the smaller subtree.`

9   Choose $b \in \{0,1\}$ with subtree rooted at v_b no larger than one rooted at v_{1-b}.

10 $\mathcal{Y} \leftarrow \left\{ y \in \mathcal{Y} \mid \oplus_{(i,j) \in S} y_{i,j} = b \right\}$

11 $(\mathcal{E}, (i^*, j^*)) \leftarrow$ **row-reduce**$(\mathcal{E}, D_\mathcal{Y}, \oplus_{(i,j) \in S} y_{i,j} = b)$

12   **add** (i^*, j^*) to $D_\mathcal{Y}$

13 $\mathcal{X} \leftarrow \mathcal{X}\,|_{x_{i^*} \neq j^*}$ `// Note:  if` $i^* \in I$ `then` $x_{i^*} \neq j^*$ `already.`

14 $v \leftarrow v_b$

15   **end**

16   **end**

17   **if** *Alice speaks at v and partitions $\mathcal{X}$ into $\mathcal{X}^0 \cup \mathcal{X}^1$* **then**

18   Let $b \in \{0,1\}$ be such that $|\mathcal{X}^b| \geq \frac{1}{2} \cdot |\mathcal{X}|$

19 $\mathcal{X} \leftarrow \mathcal{X}^b$

20 $v \leftarrow v_b$

21   **end**

22   **if** *min-entropy-rate$(\mathcal{X}) < \tau = 1/2$* **then**

23   **RestoreMinEntropyRateAndQuery**$(\mathcal{X}, z)$

24   **end**

25 end

26 return *label of v*

Algorithm 1: Simulation algorithm

1 Algorithm: RestoreMinEntropyRateAndQuery($\mathcal{X}, z$)

Data: $\mathcal{X} \subseteq [m]^N, z \in \{0,1\}^N$.

Result: Updates $\mathcal{X}$ to restore min-entropy rate to $\tau = 1/2$ by fixing coordinates via queries
to z.

2 Let $I' \subseteq [N] \setminus I$ be a maximal set on on which $\mathcal{X}$ has min-entropy rate $< \tau = 1/2$

3 Let $\alpha_{I'} \in [m]^{I'}$ be such that $\Pr_{x \in \mathcal{X}}[x_{I'} = \alpha_{I'}] > m^{-\tau|I'|}$

4 $\mathcal{X} \leftarrow \{x \in \mathcal{X} \mid x_{I'} = \alpha_{I'}\}$

5 Query all coordinates in I' and let $z_{I'}$ be the query answers

6 $\mathcal{Y} \leftarrow \left\{y \in \mathcal{Y} \mid y_{(I', \alpha_{I'})} = z_{I'}\right\}$

7 foreach $i \in I'$ **do**

8 $\rho(i) \leftarrow z_i$

9 $(\mathcal{E}, (i^*, j^*)) \leftarrow$ **row-reduce**$(\mathcal{E}, D_{\mathcal{Y}}, y_{i, \alpha_i} = \rho(i))$ `// Note:` $i^* = i, \ j^* = \alpha_i$

10   **add** (i, α_i) to $D_{\mathcal{Y}}$

11 end

12 $I \leftarrow I \cup I'$

Algorithm 2: Procedure **RestoreMinEntropyRateAndQuery**

4.3 Analysis of the simulation algorithm

We first argue that $\mathcal{X}$ and $\mathcal{Y}$ are never empty during the run of our simulation algorithm. Thus, when the algorithm reaches a leaf node of Π we can output a correct answer. To do so we observe the following invariants on our simulation algorithm (algorithm 1).

Lemma 4.2 (Invariants of the Simulation Algorithm (algorithm 1)). *At the beginning of every iteration of the while loop in Algorithm 1, the following properties hold:*

    a. ρ defines the path in the decision tree T that is the outcome of the queries and fixed(ρ) = I.

    b. $\mathcal{Y}$ is the set of inputs satisfying $\mathcal{E}$ which is row-reduced on $D_{\mathcal{Y}}$.

    c. For any $x \in \mathcal{X}$, and any $i \notin I$, (i, x_i) is a free coordinate of $\mathcal{Y}$; that is $(i, x_i) \notin D_{\mathcal{Y}}$.

    d. For every total assignment to the free coordinates $([N] \times [m]) \setminus D_{\mathcal{Y}}$, there is a unique assignment to $D_{\mathcal{Y}}$ that extends it to an element of $\mathcal{Y}$.

    e. $\mathcal{X}$ has min-entropy rate at least τ on $[N] \setminus I$

Proof. All but the last of the conditions of Lemma 4.2 easily can be seen to hold by inspection of Algorithms 1 and 2. The last follows by the argument of GPW and follows from the maximality of the set I' in Algorithm 2. It is easy to see that if $\mathcal{X}$ has non-zero min-entropy rate, then $\mathcal{X}$ is non-empty. Moreover, since every element of $\mathcal{X}$ points only to free coordinates in blocks outside of I and the min-entropy on each block is large, $\mathcal{Y}$ must have many free coordinates outside of I. $\qquad\square$

We bound the number of queries $|I|$ by using a potential function equal to the deficiency of $\mathcal{X}_{[N]\setminus I}$. Let A be the number of bits spoken by Alice, and B be the number of bits spoken by Bob in Π. We analyze the change in $D_{\infty}(\mathcal{X}_{[N]\setminus I})$ due to updates of $\mathcal{X}$ and I:

13

- line 13 in Algorithm 1: removing $x_{i^*} = j^*$ from $\mathcal{X}$ for newly dependent (i^*, j^*):
  By Lemma 4.2(e), we know that $\mathcal{X}$ has min-entropy rate at least $\tau = 1/2$. Thus, $\Pr_{x \sim \mathcal{X}}[x_{i^*} = j^*] \leq 1/m^\tau \leq 1/2$ since $m \geq 4$ and $\tau = 1/2$. Consequently, $\Pr_{x \sim \mathcal{X}}[x_{i^*} \neq j^*] \geq 1/2$. So, by Fact 2.3, $H_\infty(\mathcal{X}\mid_{x_{i^*} \neq j^*}) \geq H_\infty(\mathcal{X}) - 1$. Therefore, the change in $D_\infty(\mathcal{X}_{[N] \setminus I})$ is at most 1. This step is executed at most B times. (Note: For larger m we could maintain sharper bounds, but it seems that we don't need to do so.)

- line 19 in Algorithm 1: choosing the more frequent bit of Alice to send:
  This increases $D_\infty(\mathcal{X}_{[N] \setminus I})$ by at most 1. This step is executed A times.

- line 4 and line 10 in Algorithm 2: querying and fixing coordinates I' maximal for min-entropy loss:
  First, line 4 increases $D_\infty(\mathcal{X}_{[N] \setminus I})$ by at most $\tau \cdot |I'| \cdot \log_2 m$ as shown the proof of Lemma 3.5 in [13]. Second, line 10 decreases $D_\infty(\mathcal{X}_{[N] \setminus I})$ by precisely $|I'| \cdot \log_2 m$ since it adds $|I'|$ blocks to I. The net total of these changes is that $D_\infty(\mathcal{X}_{[N] \setminus I})$ decreases by at least $(1 - \tau) \cdot |I'| \cdot \log_2 m$ in this case.

Putting these together yields:

$$D_\infty(\mathcal{X}_{[N] \setminus I}) \leq A + B - (1 - \tau) \cdot |I| \cdot \log_2 m.$$

Since $D_\infty(\mathcal{X}_{[N] \setminus I}) \geq 0$ we must have

$$|I| \leq \frac{A + B}{(1 - \tau) \log_2 m}.$$

Since $\tau = 1/2$, $A + B \geq 0.5 |I| \log_2 m$ and hence $C^{*,\oplus}(R \circ \mathrm{IND}_m^N) \geq A + B \geq 0.5\, C^{dt}(R) \log_2 m$. □

We can strengthen the above in the case that each of Alice's bits, like Bob's bits, either is irrelevant or splits $\mathcal{X}$ exactly in half. Then, as with our simulation of Bob's bits, we choose to follow the side with the smaller protocol subtree. We see that the paths followed in the protocol Π are of total length (in bits that matter to the simulation) at most the logarithm of the size of Π.

In particular, this applies if m is a power of 2 so that each x_i is represented by a series of bits, Alice's bits are also parities, and we replace line 13 by constraining one bit of x_{i^*} that isn't already constrained to be different from the corresponding bit of j^*. We write $L^{\oplus,\oplus}(R)$ for the number of leaves (i.e., the size) of a protocol Π for R in which both Alice and Bob only send parities, which we call *parity communication*. Using this obtain the following:

Theorem 4.3. $C^{dt}(R)$ is $O(\log_m(L^{\oplus,\oplus}(R \circ IND_m^N)))$

4.4 Parity decision trees and Res($\oplus$) proofs

We can use the same ideas with small modifications to give a generic method for producing lower bounds for parity decision trees from those for ordinary decision trees.

Theorem 4.4. *For any sufficiently large m that is a power of 2 and any function $f : \{0,1\}^N \to \{0,1\}$, $C_\oplus^{dt}(f \circ IND_m^N) \geq C^{dt}(f)$ and $\text{size}_\oplus^{dt}(f \circ IND_m^N) \geq 2^{C^{dt}(f)} \geq \text{size}^{dt}(f)$.*

Proof. We follow the ideas of the proof of Theorem 4.3 with a small modification that combines the steps for Bob and for Alice as follows: We maintain $\mathcal{X}$ as an affine subspace as before and we maintain a set $D_{\mathcal{Y}} \subset [N] \times [m]$ of coordinates as before (though $\mathcal{Y}$ itself is not maintained), but now the equations that we maintain involving these coordinates depend on the bits of the x_i also. At each parity that is not already implied, we row-reduce to remove the variables in $D_{\mathcal{Y}}$.

If a coordinate (i^*, j^*) in $\mathcal{Y}$ remains, we choose it as the new dependent coordinate and complete the row-reduction. We then apply the portion of the simulation designated as Bob's simulation in order to ensure that x_{i^*} does not point to j^*. If no such coordinate remains, then this becomes a constraint on $\mathcal{X}$ and we apply the portion associated with Alice.

In either case, we add one defining equation for $\mathcal{X}$ for each bit that is sent that is not already implied, so the deficiency of $\mathcal{X}$ is precisely this number. The same analysis shows that $C^{dt}(f)$ is $O(\log_m(\text{size}^{dt}_\oplus(f \circ \text{IND}^N_m)))$ and hence $O(C^{dt}_\oplus(f \circ \text{IND}^N_m)/\log m)$. By choosing m a sufficiently large constant, we obtain that $C^{dt}(f) \leq \log_2(\text{size}^{dt}_\oplus(f \circ \text{IND}^N_m))$ and hence $\text{size}^{dt}(f) \leq 2^{C^{dt}(f)} \leq \text{size}^{dt}_\oplus(f \circ \text{IND}^N_m)$ and $C^{dt}(f) \leq C^{dt}_\oplus(f \circ \text{IND}^N_m)$. $\qquad\square$

We can use this (in the form for relations, which we could easily have stated above) to show that we can convert each k-CNF φ to an $O(k)$-CNF $\varphi \circ \text{IND}^N_m$ that requires tree-like $Res(\oplus)$ refutations for $\varphi \circ \text{IND}^N_m$ that are at least linear in the size of tree-like resolution refutations for φ.

Corollary 4.5. *For any sufficiently large integer ℓ, $m = 2^\ell$, and any unsatisfiable k-CNF formula with M clauses on N Boolean variables, $\varphi \circ IND^N_m$ is a $k(\ell + 1)$-CNF formula with $M' = m^k M$ clauses on $N' = N(\ell + m)$ variables that requires $\text{tree}_{Res(\oplus)}(\varphi \circ IND^N_m) \geq \text{tree}_{Res}(\varphi) \geq 2^{\text{width}_{Res}(\varphi)-k}$.*

Proof. By Proposition 2.8, for any minimal tree-$Res(\oplus)$ refutation of $\varphi \circ \text{IND}^N_m$ there is an isomorphic parity decision tree solving the search problem $\text{Search}_{\varphi \circ \text{IND}^N_m}$. Using the form of Theorem 4.4 for relations we can convert a parity communication protocol for $\text{Search}_{\varphi \circ \text{IND}^N_m}$ into one of at most the same size that solves the search problem Search_φ, since each correct output of $\text{Search}_{\varphi \circ \text{IND}^N_m}$ yields the name of a violated clause of $\varphi \circ \text{IND}^N_m$ that corresponds to a unique clause of φ and can be output by the ordinary decision tree. The final result follows using the equivalence of decision trees for Search_φ and tree-resolution refutations of φ from Proposition 2.7 and the tree-size/width relationship from Proposition 2.6. $\qquad\square$

We note that Itsykson and Kojevnikov [16, 17] previously used a much more specialized lifting theorem from [2] for the specific case of Tseitin formulas to give tree-like $Res(\oplus)$ lower bounds. Our new simple method is much more general and yields a large class of hard formulas. We also note that Huynh and Nordström [15] gave lifting theorems for a variety of other proof systems using constant-size index gadgets (indeed with $m = 3$) but these only yield good bounds for a restricted class of formulas whose search problems have high "critical block sensitivity".

5   Summary and future directions

Our results show that the Index (or Inner-Product) gadget is not a good disperser for low min-entropy deficiency rectangles $\mathcal{X} \times \mathcal{Y}$ when m is much smaller than $\log N$. Thus to reduce the gadget size beyond logarithmic using current techniques we need to consider other properties on the rectangle $\mathcal{X} \times \mathcal{Y}$ maintained during the simulation that can ensure that Index is a good

disperser. Our counterexample to the conjecture of Lovett et al. [20] suggests the following natural property for $\mathcal{X}, \mathcal{Y}$ in addition to having low entropy deficiency:

- Except for a small subset of blocks J of size $O(\Delta)$, every block of every y in $\mathcal{Y}$ is "almost" balanced in terms of the number of zeroes and ones. That is, for any $i \in [N] \setminus J$ and for any $y \in \mathcal{Y}$, $|y^i|_1 \approx m - |y^i|_1$, where $|y^i|_1$ denotes the number of ones in the i-th block of y.

Note that our counterexample is avoided by $\mathcal{Y}$ satisfying this property; indeed this property is violated in the extreme by our counterexample as every y in the set $\mathcal{Y}$ we construct has $\omega(\Delta)$ blocks that are maximally unbalanced.

The standard simulation paradigm allows considerable flexibility in choosing which subset to focus on in the rectangle of inputs associated with each node of the communication protocol. Maintaining something like the property above is easy to do and it is plausible that this or related properties will indeed be sufficient to yield general lifting theorems with very small Index gadgets.

Although our counterexample ruled out improving the gadget size to constant for Index in deterministic lifting theorems using current techniques, we were able to prove a lifting theorem with constant-sized gadgets for the restricted class of protocols where Bob is restricted to sending parities. A natural extension of this direction is to consider semi-structured protocols where Bob is restricted to sending other interesting functions of his input bits. A natural class of restricted functions are threshold functions. The ideas used in our lifting theorem do not work in this case. As illustrated by the application of our semi-structured lifting theorem to $\mathrm{Res}(\oplus)$ lower bounds, such restricted lifting theorems may have immediate applications in proof complexity. They may also be a natural avenue for developing new tools and techniques that could potentially help in proving general lifting theorems with constant sized gadgets.

References

[1] Ryan Alweiss, Shachar Lovett, Kewen Wu, and Jiapeng Zhang. Improved bounds for the sunflower lemma. *Annals of Mathematics*, 194(3):795–815, 2021.

[2] Paul Beame, Toniann Pitassi, and Nathan Segerlind. Lower bounds for Lovász-Schrijver systems and beyond follow from multiparty communication complexity. *SIAM J. Comput.*, 37(3):845–869, 2007.

[3] Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow - resolution made simple. *J. ACM*, 48(2):149–169, 2001.

[4] Arkadev Chattopadhyay, Yuval Filmus, Sajin Koroth, Or Meir, and Toniann Pitassi. Query-to-communication lifting using low-discrepancy gadgets. *SIAM J. Comput.*, 50(1):171–210, 2021.

[5] Arkadev Chattopadhyay, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Simulation theorems via pseudo-random properties. *Comput. Complex.*, 28(4):617–659, 2019.

[6] Arkadev Chattopadhyay, Nikhil S. Mande, Swagato Sanyal, and Suhail Sherif. Lifting to parity decision trees via stifling. 2022. To appear at ITCS 2023.

[7] Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *J. Symb. Log.*, 44(1):36–50, 1979.

[8] Ankit Garg, Mika Göös, Pritish Kamath, and Dmitry Sokolov. Monotone circuit lower bounds from resolution. *Theory Comput.*, 16:1–30, 2020.

[9] Mika Göös, Rahul Jain, and Thomas Watson. Extension complexity of independent set polytopes. *SIAM J. Comput.*, 47(1):241–269, 2018.

[10] Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. *SIAM J. Comput.*, 45(5):1835–1869, 2016.

[11] Mika Göös and Toniann Pitassi. Communication lower bounds via critical block sensitivity. *SIAM J. Comput.*, 47(5):1778–1806, 2018.

[12] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. *SIAM J. Comput.*, 47(6):2435–2450, 2018.

[13] Mika Göös, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for BPP. *SIAM J. Comput.*, 49(4), 2020.

[14] Danny Harnik and Ran Raz. Higher lower bounds on monotone size. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA*, pages 378–387. ACM, 2000.

[15] Trinh Huynh and Jakob Nordström. On the virtue of succinct proofs: amplifying communication complexity hardness to time-space trade-offs in proof complexity. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 233–248. ACM, 2012.

[16] Dmitry Itsykson and Dmitry Sokolov. Lower bounds for splittings by linear combinations. In *Mathematical Foundations of Computer Science 2014 - 39th International Symposium, MFCS 2014, Budapest, Hungary, August 25-29, 2014. Proceedings, Part II*, volume 8635 of *Lecture Notes in Computer Science*, pages 372–383. Springer, 2014.

[17] Dmitry Itsykson and Dmitry Sokolov. Resolution over linear equations modulo two. *Ann. Pure Appl. Log.*, 171(1), 2020.

[18] R. Kaas and J. M. Buhrman. Mean, median and mode in binomial distributions. *Statistica Neerlandica*, 34(1):13–18, 1980.

[19] James R. Lee, Prasad Raghavendra, and David Steurer. Lower bounds on the size of semidefinite programming relaxations. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 567–576. ACM, 2015.

[20] Shachar Lovett, Raghu Meka, Ian Mertz, Toniann Pitassi, and Jiapeng Zhang. Lifting with sunflowers. In *13th Innovations in Theoretical Computer Science Conference, ITCS 2022, January*

31 - February 3, 2022, Berkeley, CA, USA, volume 215 of *LIPIcs*, pages 104:1–104:24. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

[21] Ran Raz and Pierre McKenzie. Separation of the monotone NC hierarchy. In *38th Annual Symposium on Foundations of Computer Science, FOCS '97, Miami Beach, Florida, USA, October 19-22, 1997*, pages 234–243. IEEE Computer Society, 1997.

[22] Ran Raz and Iddo Tzameret. Resolution over linear equations and multilinear proofs. *Ann. Pure Appl. Log.*, 155(3):194–224, 2008.

[23] Robert Robere, Toniann Pitassi, Benjamin Rossman, and Stephen A. Cook. Exponential lower bounds for monotone span programs. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, New Brunswick, New Jersey, USA*, pages 406–415. IEEE Computer Society, 2016.

[24] Alexander A. Sherstov. The pattern matrix method. *SIAM J. Comput.*, 40(6):1969–2000, 2011.

[25] Xiaodi Wu, Penghui Yao, and Henry S. Yuen. Raz-McKenzie simulation with the inner product gadget. *Electron. Colloquium Comput. Complex.*, TR17-010, 2017.