Communication Lower Bounds via Critical Block Sensitivity

[Extended Abstract]*

Mika Göös
Department of Computer Science
University of Toronto
mgoos@cs.toronto.edu

Toniann Pitassi
Department of Computer Science
University of Toronto
toni@cs.toronto.edu

ABSTRACT

We use *critical block sensitivity*, a new complexity measure introduced by Huynh and Nordström (STOC 2012), to study the communication complexity of search problems. To begin, we give a simple new proof of the following central result of Huynh and Nordström: if S is a search problem with critical block sensitivity b, then every randomised two-party protocol solving a certain *two-party lift* of S requires $\Omega(b)$ bits of communication. Besides simplicity, our proof has the advantage of generalising to the multi-party setting. We combine these results with new critical block sensitivity lower bounds for *Tseitin* and *Pebbling* search problems to obtain the following applications.

- **Monotone circuit depth:** We exhibit a monotone function on n variables whose monotone circuits require depth $\Omega(n/\log n)$; previously, a bound of $\Omega(\sqrt{n})$ was known (Raz and Wigderson, JACM 1992). Moreover, we prove a tight $\Theta(\sqrt{n})$ monotone depth bound for a function in monotone P. This implies an average-case hierarchy theorem within monotone P similar to a result of Filmus et al. (FOCS 2013).
- **Proof complexity:** We prove new rank lower bounds as well as obtain the first length–space lower bounds for semi-algebraic proof systems, including Lovász–Schrijver and Lasserre (SOS) systems. In particular, these results extend and simplify the works of Beame et al. (SICOMP 2007) and Huynh and Nordström.

Categories and Subject Descriptors

F.0 [**Theory of Computation**]: General; F.1.3 [**Computation by Abstract Devices**]: Complexity Measures and Classes.

Keywords

Communication complexity, Monotone circuit depth, Proof complexity

*The full version of this paper, containing most of the proofs, is available at arXiv:1311.2355 [24].

STOC'14, May 31 – June 03 2014, New York, NY, USA

http://dx.doi.org/10.1145/2591796.2591838

1. INTRODUCTION

Apart from their intrinsic interest, communication lower bounds for *search problems* find applications in two major areas of complexity theory.

1. **Circuit complexity:** A famous theorem of Karchmer and Wigderson [31] states that for all boolean functions f, the minimum depth of a circuit computing f is equal to the communication complexity of a certain search problem, called the *Karchmer–Wigderson (KW) game* for f. While it still remains a major open problem to prove general depth lower bounds for explicit boolean functions, KW-games have permitted progress in *monotone* circuit complexity: there are monotone depth lower bounds for graph connectivity [31], clique functions [22, 42], perfect matchings [42], and functions in monotone P [40]. See also Chapter 7 in Jukna's new book [29].
2. **Proof complexity:** Impagliazzo et al. [28] (see also [29, §19.3]) introduced an analogue of KW-games to proof complexity. They showed how small tree-like Cutting Planes refutations of an unsatisfiable CNF formula F can be converted into efficient *two-party* communication protocols for a certain canonical search problem associated with F. More recently, Beame et al. [5] extended this connection by showing that suitable lower bounds for *multi-party* protocols imply degree/rank lower bounds for many well-studied semi-algebraic proof systems, including Lovász–Schrijver [35], Positivstellensatz [25], Sherali–Adams [46], and Lasserre (SOS) [33] systems. In parallel to these developments, Huynh and Nordström [27] have also found a new kind of simulation of space-bounded proofs by communication protocols. They used this connection to prove length–space lower bounds in proof complexity.

In this work we obtain new randomised lower bounds for search problems in both two-party and multi-party settings. Our proofs are relatively simple reductions from the *set-disjointness* function, the canonical NP-complete problem in communication complexity. These results allow us to derive, almost for free, new lower bounds in the above two application domains.

1. **Monotone depth:** We construct a monotone function on n variables whose monotone circuits require depth $\Omega(n/\log n)$. Previously, the best bound for an explicit monotone function (perfect matchings) was $\Omega(\sqrt{n})$ due to Raz and Wigderson [42]. Moreover, we prove a tight $\Theta(\sqrt{n})$ monotone depth bound for a function in monotone P.

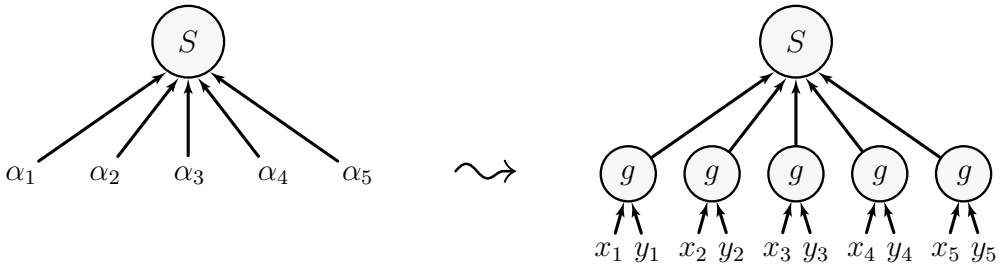

Figure 1: Composing a search problem S with a two-party gadget g.

In fact, we show that the above bounds hold even if the circuits are allowed to make some errors. In particular, we get a simple proof of an average-case hierarchy theorem within monotone P, similar to a recent result of Filmus et al. [18]. (Their result was proven using Fourier analytic techniques [39, 12].)

2. **Rank, length, and space:** We obtain new rank lower bounds for a family of semantic polynomial threshold proof systems called $\mathsf{T}^{\mathsf{cc}}(k)$, which includes many of the semi-algebraic proof systems mentioned above. This extends and simplifies the work of Beame et al [5]. We also extend the length–space lower bound of Huynh and Nordström [27] to hold for $\mathsf{T}^{\mathsf{cc}}(k)$ systems of degree up to $k = (\log n)^{1-o(1)}$. In particular, this yields the first nontrivial length–space lower bounds for dynamic SOS proofs of this degree.

We state these results more precisely shortly, once we first formalise our basic communication complexity setup.

1.1 Starting point: Critical block sensitivity

We build on the techniques recently introduced by Huynh and Nordström [27]. They defined a new complexity measure for search problems called *critical block sensitivity*, which is a generalisation of the usual notion of block sensitivity for functions (see [10] for a survey). They used this measure to give a general method of proving lower bounds for *composed* search problems in the two-party communication model. These notions will be so central to our work that we proceed to define them immediately.

A *search problem* on n variables is a relation $S \subseteq \{0,1\}^n \times Q$ where Q is some set of possible solutions. On input $\alpha \in \{0,1\}^n$ the search problem is to find a solution $q \in Q$ that is *feasible for* α, that is, $(\alpha, q) \in S$. We assume that S is such that all inputs have at least one feasible solution. An input is called *critical* if it has a unique feasible solution.

Definition 1 (Critical block sensitivity [27]). *Fix a search problem $S \subseteq \{0,1\}^n \times Q$. Let $f \subseteq S$ denote a total function that solves S, i.e., for each input $\alpha \in \{0,1\}^n$ the function picks out some feasible solution $f(\alpha)$ for α. We denote by* $\mathrm{bs}(f,\alpha)$ *the usual block sensitivity of f at α. That is,* $\mathrm{bs}(f,\alpha)$ *is the maximal number* bs *such that there are disjoint blocks of coordinates $B_1, \dots, B_{\mathrm{bs}} \subseteq [n]$ satisfying $f(\alpha) \neq f(\alpha^{B_i})$ for all i; here, α^{B_i} is the same as α except the input bits in coordinates B_i are flipped. The* critical block sensitivity *of S is defined as*

$$\mathrm{bs}_{crit}(S) \;:=\; \min_{f \subseteq S} \max_{\text{critical } \alpha} \mathrm{bs}(f,\alpha).$$

We note immediately that $\mathrm{bs}_{crit}(S)$ is a lower bound on the deterministic decision tree complexity of S. Indeed, a deterministic decision tree defines a total function $f \subseteq S$ and on each critical input α the tree must query at least one variable from each sensitive block of f at α (see [10, Theorem 9]). It turns out that $\mathrm{bs}_{crit}(S)$ is also a lower bound on the *randomised* decision tree complexity (see Theorem 1 below).

1.2 Composed search problems

In order to study a search problem $S \subseteq \{0,1\}^n \times Q$ in the setting of two-party communication complexity, we need to specify how the n input variables of S are divided between the two players, Alice and Bob.

Unfortunately, for many search problems (and functions) there is often no partition of the variables that would carry the "intrinsic" complexity of S over to communication complexity. For example, consider computing the AND function on n inputs. The block sensitivity of AND is n, but this complexity is lost once we move to the two-party setting: only $O(1)$ many bits need to be communicated between Alice and Bob regardless of the input partition.

For this reason, one usually studies *composed* (or *lifted*) variants $S \circ g^n$ of the original problem; see Figure 1. In a composed problem, each of the n input bits of S are encoded using a small two-party function $g\colon \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, sometimes called a *gadget*. As input to $S \circ g^n$ Alice gets an $x \in \mathcal{X}^n$ and Bob gets a $y \in \mathcal{Y}^n$. We think of the pair (x,y) as encoding the input

$$\alpha = g^n(x,y) = (\, g(x_1,y_1), \dots, g(x_n,y_n)\,)$$

of the original problem S. The objective is to find a $q \in Q$ such that $(g^n(x,y), q) \in S$.

1.3 Our communication complexity results

We start by giving a simple new proof of the following central result of Huynh and Nordström [27]. (Strictly speaking, the statement of the original theorem [27] is slightly weaker in that it involves an additional "consistency" assumption, which we do not need.)

Theorem 1 (Two-party version). *There is a two-party gadget $g\colon \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ such that if $S \subseteq \{0,1\}^n \times Q$ is any search problem, then $S \circ g^n$ has randomised bounded-error communication complexity $\Omega(\mathrm{bs}_{crit}(S))$.*

Huynh and Nordström proved Theorem 1 for the gadget $g = \textsc{3ind}$, where $\textsc{3ind}\colon [3] \times \{0,1\}^3 \to \{0,1\}$ is the indexing function that maps $(x,y) \mapsto y_x$. Their proof used the information complexity approach [11, 2] and is quite intricate.

By contrast, we prove Theorem 1 by a direct randomised reduction from the *set-disjointness* function

$$\mathrm{DISJ}_n(x,y) = (\mathrm{OR}_n \circ \mathrm{AND}^n)(x,y) = \bigvee_{i\in[n]} (x_i \wedge y_i).$$

In the language of Babai et al. [1] (see also [15]) the set-disjointness function is NP-complete in communication complexity: it is easy to certify that $\mathrm{DISJ}_n(x,y) = 1$, and conversely, every two-party function with low nondeterministic complexity reduces efficiently to DISJ_n. Our proof of Theorem 1 is inspired by a result of Zhang [51] that essentially establishes Theorem 1 in case S is a function and $\mathrm{bs}_{crit}(S)$ is simply the standard block sensitivity. The new key insight in our proof is the following.

> **Key idea:** We choose g to be *random-self-reducible*. (see Section 2 for definitions.)

Random-self-reducibility is a notion often studied in cryptography and classical complexity theory, but less often in communication complexity. Most notably, random-self-reducibility was used implicitly in [42]. The definitions we adopt are similar to those introduced by Feige et al. [17] in a cryptographic context.

Our proof has also the advantage of generalising naturally to the multi-party setting. This time we start with the k-party unique-disjointness function $\mathrm{UDISJ}_{k,n}$ and the proof involves the construction of k-party random-self-reducible functions g_k.

THEOREM 2 (MULTI-PARTY VERSION). *There are k-party gadgets $g_k\colon \mathcal{X}^k \to \{0,1\}$ with domain size $\log|\mathcal{X}| = k^{o(1)}$ bits per player, such that if $S \subseteq \{0,1\}^n \times Q$ is any search problem, then $S \circ g_k^n$ has randomised bounded-error communication complexity at least that of $\mathrm{UDISJ}_{k,\mathrm{bs}}$ (up to constants), where $\mathrm{bs} = \mathrm{bs}_{crit}(S)$.*

Theorem 2 can be applied to the following multi-player communication models.

- **Number-in-hand:** The i-th player only sees the i-th part of the input. Here, set-disjointness has been studied under broadcast communication (e.g., [26]) and under private channel communication [9].
- **Number-on-forehead (NOF):** The i-th player sees all parts of the input except the i-th part [13]. The current best randomised lower bound for $\mathrm{UDISJ}_{k,n}$ is $\Omega(\sqrt{n}/2^k k)$ by Sherstov [48]. We rely heavily on Sherstov's result in our proof complexity applications.

In the rest of this introduction we discuss the applications—the impatient reader who wants to see the proof of Theorem 1 can immediately skip to Sections 2 and 3.

1.4 CSPs and their canonical search problems

To get the most out of Theorems 1 and 2 for the purposes of applications, we need to find search problems with high critical block sensitivity but low certificate complexity. Low-degree constraint satisfaction problems (CSPs) capture exactly the latter goal [34].

DEFINITION 2 (d-CSPs). *A CSP F consists of a set of (boolean) variables $vars(F)$ and a set of constraints $cons(F)$. Each constraint $C \in cons(F)$ is a function that maps a truth assignment $\alpha\colon vars(F) \to \{0,1\}$ to either 0 or 1. If $C(\alpha) = 1$, we say that C is* satisfied *by α, otherwise C is* violated *by α. Let $vars(C)$ denote the smallest subset of $vars(F)$ such that C depends only on the truth values of the variables in $vars(C)$. We say that F is of* degree d*, or F is a d-CSP, if $|vars(C)| \le d$ for all C. Note that d-CNF formulas are a special case of d-CSPs, and conversely, each d-CSP can be written as an equivalent d-CNF with a factor 2^d blow-up in the number of constraints.*

An *unsatisfiable* CSP F has no assignment that satisfies all the constraints. Each such F comes with an associated *canonical search problem* $S(F)$.

DEFINITION 3 (CANONICAL SEARCH PROBLEMS). *Let F be an unsatisfiable CSP. In the search problem $S(F)$ we are given an assignment $\alpha\colon vars(F) \to \{0,1\}$ and the goal is to find a constraint $C \in cons(F)$ that is violated by α.*

We give new critical block sensitivity lower bounds for the canonical search problems associated with *Tseitin* and *Pebbling* formulas.

1.5 Sensitivity of Tseitin formulas

Tseitin formulas are well-studied examples of unsatisfiable CSPs that are hard to refute in many proof systems; for an overview, see Jukna [29, §18.7].

DEFINITION 4 (TSEITIN FORMULAS). *Let $G = (V, E, \ell)$ be a connected labelled graph of maximum degree d where the labelling $\ell\colon V \to \{0,1\}$ has odd Hamming weight. The* Tseitin formula Tse_G *associated with G is the d-CSP that has the edges $e \in E$ as variables and for each node $v \in V$ there is a constraint C_v defined by*

$$C_v(\alpha) = 1 \quad \Longleftrightarrow \quad \sum_{e:v\in e} \alpha(e) \equiv \ell(v) \pmod 2.$$

It follows from a simple parity argument that Tse_G is unsatisfiable.

Call G *κ-routable* if there is a set $T \subseteq V$ of size $|T| \ge 2\kappa$ such that for any set of κ disjoint pairs of nodes of T there are κ edge-disjoint paths in G that connect all the pairs. (Warning: κ-routability is usually defined only for $T = V$, but we relax this condition.) The proof of the following theorem appears in the full version [24].

THEOREM 3 (TSEITIN SENSITIVITY). *If G is κ-routable, then $\mathrm{bs}_{crit}(S(Tse_G)) = \Omega(\kappa)$.*

Theorem 3 can be applied to the following classes of bounded-degree graphs.

- **Grid graphs:** If G is a $\sqrt{n}\times\sqrt{n}$ grid graph, then we can take $\kappa = \Omega(\sqrt{n})$ by letting $T \subseteq V$ be any row (or column) of nodes. This is tight: the deterministic decision tree that solves $S(Tse_G)$ using binary search makes $O(\sqrt{n})$ queries.
- **Expanders:** If G is a sufficiently strong expander (e.g., a Ramanujan graph [36]), then we can take $\kappa = \Omega(n/\log n)$ as shown by Frieze et al. [20, 19].
- **Connectors:** A *κ-connector* is a bounded-degree graph with κ inputs $I \subseteq V$ and κ outputs $O \subseteq V$ such that for any one-to-one correspondence $\pi\colon I \to O$ there exist κ edge-disjoint paths that connect $i \in I$ to $\pi(i) \in O$. If

we merge I and O in a 2κ-connector in some one-to-one manner and let $T = I = O$, we get a κ-routable graph. Conversely, if G is κ-routable, we can partition the set T as $I \cup O$ and get a κ-connector.

It is known that simple κ-connectors with $\kappa = \Theta(n/\log n)$ exist and this bound is the best possible [38]. Thus, the best lower bound provable using Theorem 3 is $\Theta(n/\log n)$.

It is well known that the *deterministic* decision tree complexity of $S(Tse_G)$ is $\Omega(n)$ when G is an expander [50]. However, *randomised* lower bounds—which Theorem 3 provides—are more scarce. We are only aware of a single previous result in the direction of Theorem 3, namely, Lovász et al. [34, §3.2.1] announce a lower bound of $\Omega(n^{1/3})$ for the randomised decision tree complexity of $S(Tse_G)$ when G is an expander. Our Theorem 3 subsumes this.

1.6 Sensitivity of pebbling formulas

Pebble games have been studied extensively as means to understand time and space in computations; for an overview, see the survey by Nordström [37]. In this work we restrict our attention to the simple *(black) pebble game* that is played on a directed acyclic graph G with a unique sink node t (i.e., having outdegree 0). In this game the goal is to place a pebble on the sink t using a sequence of *pebbling moves*. The allowed moves are:

(1) A pebble can be placed on a node if its in-neighbours have pebbles on them. In particular, we can always pebble a source node (i.e., having indegree 0).
(2) A pebble can be removed from any pebbled node (and reused later in the game).

The *(black) pebbling number* of G is the minimum number of pebbles that are needed to pebble the sink node in the pebble game on G.

The pebble game on G comes with an associated *pebbling formula*; see [8] and [37, §2.3].

Definition 5 (Pebbling formulas).
Let $G = (V, E, t)$ be a directed acyclic graph of maximum indegree d where t is a unique sink. The pebbling formula *Peb_G associated with G is the $(d+1)$-CSP that has the nodes $v \in V$ as variables and the following constraints:*

(1) The variable corresponding to the sink t is false.
(2) For all nodes v with in-neighbours $w_1, \ldots, w_d$, we require that if all of $w_1, \ldots, w_d$ are true, then v is true. In particular, each source node must be true.

It is not hard to see that Peb_G is unsatisfiable.

Classical complexity measures for $S(Peb_G)$ include the pebbling number of G (a measure of *space*) and the deterministic decision tree complexity (a measure of *parallel time*). However, these complexity measures are fundamentally *deterministic* and do not seem to immediately translate into *randomised* lower bounds, which are needed in our applications.

For this reason, Huynh and Nordström [27] devised an elegant ad hoc proof method for their result that, for a pyramid graph G, $\mathrm{bs}_{crit}(S(Peb_G)) = \Omega(n^{1/4})$. Annoyingly, this falls a little short of both the pebbling number $\Theta(\sqrt{n})$ of G and the decision tree complexity $\Theta(\sqrt{n})$ of $S(Peb_G)$. Here we close this gap by generalising their proof method: we get tight bounds for a different (but related) graph G. The proof appears in the full version [24].

Theorem 4 (Pebbling sensitivity).
There are bounded-degree graphs G on n nodes such that

- *G has pebbling number $\Theta(\sqrt{n})$.*
- *$S(Peb_G)$ has decision tree complexity $\Theta(\sqrt{n})$.*
- *$S(Peb_G)$ has critical block sensitivity $\Theta(\sqrt{n})$.*

1.7 Applications: Monotone depth

Raz and McKenzie [40] developed a general framework to prove monotone depth lower bounds for many monotone functions. We borrow the following piece from their machinery. Here we denote by $\mathrm{depth}_+(f)$ the minimum depth of a monotone circuit computing f.

Theorem 5 (Raz–McKenzie transformation).
Let $g\colon \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ be a two-party gadget and let F be an unsatisfiable d-CSP on n variables and m constraints. There is an explicit construction of a monotone function $f\colon \{0,1\}^N \to \{0,1\}$ on $N = m|\mathcal{X}|^d$ inputs s.t. $\mathrm{depth}_+(f)$ is lower bounded by the (deterministic) communication complexity of $S(F) \circ g^n$.

Raz and McKenzie proved only a special case of Theorem 5, but their proof (Lemma 3.5 in [40]) can be modified to yield the above general construction. In their original applications, Raz and McKenzie considered gadgets whose size grew polynomially with n. In our applications, we can take g to be the constant-size gadget from Theorem 1 and this way make an extremely efficient use of Theorem 5.

Monotone depth from Tseitin.

First, let G be a $\Omega(n/\log n)$-routable graph with n nodes and bounded degree $d = O(1)$. Then $S(Tse_G)$ is the canonical search problem associated with a d-CSP on $O(n)$ variables and n constraints. Theorems 1 and 3 tell us that $S(Tse_G) \circ g^n$ has two-party communication complexity $\Omega(n/\log n)$.

Corollary 6 (Monotone depth from Tseitin).
There is an explicit monotone function f on N inputs such that $\mathrm{depth}_+(f) = \Omega(N/\log N)$. □

We recall again that the best explicit bound known previously was $\Omega(N^{1/2})$ [42]. However, it should be noted that [42] considered a very natural perfect matching function, whereas our function f is rather artificial.

Monotone depth from pebbling.

Second, we note that our methods give perhaps the simplest proof yet of a dense hierarchy theorem within monotone P (in particular, separating the monotone NC hierarchy), originally proved by [40]. Indeed, if we apply Theorem 5 for the pebbling formula Peb_G given in Theorem 4, we end up with a certain function called GEN_G that was in fact the original focus of [40]. They observed that GEN_G has polynomial size monotone circuits of depth given by the decision tree complexity of $S(Peb_G)$. Their main technical contribution was proving a matching lower bound under some additional assumptions; we can now replace this lower bound by those given by Theorems 1 and 4:

Corollary 7 (Monotone depth from pebbling).
There is an explicit function f on N inputs such that f admits polynomial size monotone circuits of $\mathrm{depth}_+(f) = \Theta(N^{1/2})$. □

(The hierarchy theorem then follows by a standard padding argument [40].)

The original bounds of [40] went up to $\Omega(N^{\delta})$ for a small constant δ. This was recently improved by the works [12, 18] that prove (among other things) monotone depth bounds of up to $\Omega(N^{1/6-o(1)})$ for GEN_G type functions.

Average-case hardness.

Since our communication lower bounds are randomised, it is natural to expect that we also get average-case lower bounds for monotone circuit depth. However, it seems that a precise connection in this direction has not been formalised before. Some related results are known: Filmus et al. [18] show that the *converse* of such a connection fails in a certain distributional sense. Raz and Wigderson [41] use randomised communication lower bounds for a different purpose, namely, to prove that every sufficiently shallow circuit for a particular function requires many negated inputs.

We provide an average-case circuit-to-protocol simulation that relies fundamentally on random-self-reducibility. The proof appears in the full version [24].

THEOREM 8 (PROTOCOLS FROM AVG-CASE CIRCUITS). *Let g, F, and $f\colon \{0,1\}^N \to \{0,1\}$ be as in Theorem 5 and assume that g is random-self-reducible. There is a distribution μ on $\{0,1\}^N$ such that if $\tilde{f}\colon \{0,1\}^N \to \{0,1\}$ is any monotone function that δ-*correlates with f, *i.e.,*

$$\Pr_{\boldsymbol{z}\sim\mu}[\, f(\boldsymbol{z}) = \tilde{f}(\boldsymbol{z})\,] \geq 1/2 + \delta,$$

then there is a randomised bounded-error protocol for $S(F)\circ g^n$ of cost $O(\delta^{-1}) + \mathrm{depth}_+(\tilde{f})$.

For example, let f be the function in monotone P from Corollary 7 whose associated search problem $S(F)\circ g^n$ has randomised communication complexity $\Theta(N^{1/2})$. Then Theorem 8 tells us that every monotone function $\tilde{f}$ that $\omega(1/N^{1/2})$-correlates with f (under a certain μ_f) has $\mathrm{depth}_+(\tilde{f}) = \Omega(N^{1/2})$.

On the one hand, this is a slight improvement over a result of Filmus et al. [18] who show the existence of functions h in monotone P such that every $\tilde{h}$ that $\Omega(1/N^{1/3-\epsilon})$-correlates with h (under a certain μ_h) has $\mathrm{depth}_+(\tilde{h}) = N^{\Omega(\epsilon)}$. On the other hand, [18] also exhibit functions in monotone NC that remain hard under correlation $1/N^{\Omega(1)}$ and we are not able to match this: if we were to simply pad our function f down to NC, our correlation bounds would worsen accordingly from inverse polynomial to inverse polylogarithmic.

1.8 Applications: Proof complexity

Over the last decade or so there have been a large number of results proving lower bounds on the rank required to refute (or approximately optimise over) systems of constraints in a wide variety of semi-algebraic (a.k.a. polynomial threshold) proof systems, including Lovász–Schrijver [35], Cutting Planes [23, 16], Positivstellensatz [25], Sherali–Adams [46], and Lasserre [33] proofs. Highlights of this work include recent linear rank lower bounds for many constraint optimization problems [44, 49, 14, 45, 21]. Nearly all of these results rely on delicate constructions of local distributions that are specific to both the problem and to the proof system.

A communication complexity approach for proving lower bounds for semi-algebraic proofs was developed by Beame et al. [5]. They studied a semantic proof system called $\mathsf{T}^{\mathsf{cc}}(k)$ whose proofs consist of lines that are computed by a low-cost (i.e., polylog communication) k-party NOF protocols (see the full version [24] for definitions). They prove that if a CNF formula F has a small tree-like $\mathsf{T}^{\mathsf{cc}}(k)$ refutation, then $S(F)$ has an efficient k-party NOF protocol. Thus, lower bounds for the tree-size of $\mathsf{T}^{\mathsf{cc}}(k)$ proofs follow from NOF lower bounds for $S(F)$.

Rank lower bounds.

Using this relationship we can now prove the following result[1] for $\mathsf{T}^{\mathsf{cc}}(k)$ proof systems, where k can be almost logarithmic in the size of the formula. We state the theorem only for rank, with the understanding that a bound of $\Omega(R)$ on rank also implies a bound of $\exp(\Omega(R))$ on tree-size. The proof appears in the full version [24].

THEOREM 9 (RANK LOWER BOUNDS). *There are explicit CNF formulas F of size s and width $O(\log s)$ such that all $\mathsf{T}^{\mathsf{cc}}(k)$ refutations of F require rank at least*

$$R_k(s) = \begin{cases} s^{1-o(1)}, & \text{for } k = 2, \\ s^{1/2-o(1)}, & \text{for } 3 \leq k \leq (\log s)^{1-o(1)}. \end{cases}$$

Theorem 9 simplifies the proof of a similar theorem from [5], which held only for a specific family of formulas obtained from non-constant degree graphs, and only for $k < \log\log s$.

We note already here that the quadratic gap between $R_2(s)$ and $R_3(s)$ will be an artefact of us switching from two-party communication to multi-party communication. More specifically, while the two-party communication complexity of set-disjointness DISJ_n is $\Omega(n)$, the corresponding lower bound for three parties is only $\Omega(\sqrt{n})$ [48]. Whether the multi-party bound can be improved to $\Omega(n)$ is an open problem.

Length–space lower bounds.

Continuing in similar spirit, [27] showed how to prove length–space lower bounds for $\mathsf{T}^{\mathsf{cc}}(2)$ systems from lower bounds on the communication complexity of $S(F)$. Using this relationship together with our new multi-party lower bounds, we can extend this result to $\mathsf{T}^{\mathsf{cc}}(k)$ systems of degree $k > 2$.

THEOREM 10 (LENGTH–SPACE LOWER BOUNDS). *There are CNF formulas F of size s such that*

- *F admits a Resolution refutation of length $L = s^{1+o(1)}$ and space $Sp = s^{1/2+o(1)}$.*
- *Any length L and space Sp refutation of F in $\mathsf{T}^{\mathsf{cc}}(k)$ must satisfy*

$$Sp \cdot \log L \;\geq\; \begin{cases} s^{1/2-o(1)}, & \text{for } k = 2, \\ s^{1/4-o(1)}, & \text{for } 3 \leq k \leq (\log s)^{1-o(1)}. \end{cases}$$

We hesitate to call Theorem 10 a *tradeoff* result since our only upper bound is a refutation requiring space $Sp = s^{1/2+o(1)}$ and we do not know how to decrease this space usage by trading it for length; this is the same situation as in [27]. We also mention that while the CNF formulas F in Theorem 10 are lifted versions of pebbling formulas, we could have formulated similar length–space lower bounds for lifted

[1] Similar claims were made in [4]. Unfortunately, as pointed out by [27], Lemma 3.5 in [4] is incorrect and this renders many of the theorems in the paper incorrect.

Tseitin formulas (where, e.g., $Sp \cdot \log L \geq s^{1-o(1)}$ for $k = 2$). But for Tseitin formulas we do not have close-to-matching upper bounds.

In any case, Theorem 10 gives, in particular, the first length–space lower bounds for dynamic SOS proofs of degree k. In addition, even in the special case of $k = 2$, Theorem 10 simplifies and improves on [27]. However, for Polynomial Calculus Resolution (a $\mathsf{T}^{\mathsf{cc}}(2)$ system), the best known length–space tradeoff results are currently proved in the recent work of Beck et al. [6]. For Resolution (maybe the simplest $\mathsf{T}^{\mathsf{cc}}(2)$ system), even stronger tradeoff results have been known since [7]; see also Beame et al. [3] for nontrivial length lower bounds in the superlinear space regime. For Cutting Planes (a $\mathsf{T}^{\mathsf{cc}}(2)$ system) Theorem 10 remains the state-of-the-art to the best of our knowledge.

1.9 Models of communication complexity

We work in the standard models of two-party and multi-party communication complexity; see [32, 29] for definitions. Here we only recall some conventions about randomised protocols. A protocol Π solves a search problem S with *error* ϵ iff on any input x the probability that $(x, \Pi(x)) \in S$ is at least $1 - \epsilon$ over the random coins of the protocol. Note that $\Pi(x)$ need not be the same feasible solution; it can depend on the outcomes of the random coins. The protocol is of *bounded-error* if $\epsilon \leq 1/4$. The constant $1/4$ here can often be replaced with any other constant less than $1/2$ without affecting the definitions too much. In the case of computing boolean functions this follows from standard boosting techniques [32, Exercise 3.4]. While these boosting techniques may fail for general search problems, we do not encounter any such problems in this work.

2. VERSATILE GADGETS

In this section we introduce *versatile* two-party functions; the generalisation to multi-party functions is relegated to the full version [24]. Our proof of Theorem 1 will work whenever we choose g to be a versatile gadget.

2.1 Self-reductions and versatility

The simplest reductions between communication problems are those that can be computed without communication. Let $f_i \colon \mathcal{X}_i \times \mathcal{Y}_i \to \{0,1\}$ for $i = 1, 2$, be two-party functions. We say that f_1 *reduces to* f_2, written $f_1 \leq f_2$, if the communication matrix of f_1 appears as a submatrix of the communication matrix of f_2. Equivalently, $f_1 \leq f_2$ iff there exist one-to-one mappings π_A and π_B such that

$$f_1(x, y) = f_2(\pi_A(x), \pi_B(y)) \qquad \text{for all} \quad (x, y) \in \mathcal{X}_1 \times \mathcal{Y}_1.$$

Our restriction to one-to-one reductions above is merely a technical convenience (cf. Babai et al. [1] allow reductions to be many-to-one).

Example 1. *Let* $\mathrm{3EQ} \colon [3] \times [3] \to \{0,1\}$ *be the equality function with inputs from* $[3]$. *Then* AND *reduces to* 3EQ *since* $\mathrm{AND}(x, y) = \mathrm{3EQ}(1 + x, 3 - y)$.

We will be interested in special kinds of reductions that reduce a function to *itself*. Our first flavour of self-reducibility relates a function f and its negation $\neg f$:

Flippability. A function f is called *flippable* if $\neg f \leq f$. Note that since the associated reduction maps z-inputs to $(1 - z)$-inputs in a one-to-one fashion, a flippable function must be *balanced*: exactly half of the inputs satisfy $f(x, y) = 1$.

Example 2. *The* XOR *function is flippable:* $\neg\mathrm{XOR}(x, y) = \mathrm{XOR}(1 - x, y)$. *By contrast,* AND *and* 3EQ *are not balanced and hence not flippable.*

We will also consider randomised reductions where the two parties are allowed to *synchronise* their computations using public randomness. More precisely, even though the two parties are still not communicating, we can let the mappings π_A and π_B depend on a public random string $\boldsymbol{r} \in \{0,1\}^*$, whose distribution the two parties can freely choose. This way, a random reduction computes $(x, y) \mapsto (\pi_A(x, \boldsymbol{r}), \pi_B(y, \boldsymbol{r}))$. The following definition is similar to the *perfectly secure* functions of Feige et al. [17].

Random self-reducibility. A function f is called *random-self-reducible* if there are mappings π_A and π_B together with a random variable $\boldsymbol{r}$ such that for every z-input $(x, y) \in f^{-1}(z)$ the random pair $(\pi_A(x, \boldsymbol{r}), \pi_B(y, \boldsymbol{r}))$ is uniformly distributed among all the z-inputs of f.

Example 3. *The equality function* $\mathrm{EQ} \colon [n] \times [n] \to \{0,1\}$ *is random-self-reducible: we can use the public randomness to sample a permutation* $\boldsymbol{\pi} \colon [n] \to [n]$ *uniformly at random and let the two parties compute* $(x, y) \mapsto (\boldsymbol{\pi}(x), \boldsymbol{\pi}(y))$. *(In fact, to further save on the number of random bits used, it would suffice to choose* $\boldsymbol{\pi}$ *from any group that acts 2-transitively on* $[n]$*.)*

A notable example of a function that is *not* random-self-reducible is AND; it has only one 1-input, which forces any self-reduction to be the identity map. This is particularly inconvenient since AND is featured in the set-disjointness function $\mathrm{DISJ}_n = \mathrm{OR}_n \circ \mathrm{AND}^n$, which will be the starting point for our reductions. To compensate for the shortcomings of AND we work with a slightly larger function $g \geq \mathrm{AND}$ instead.

Definition 6 (Versatility). *A two-party function* g *is called* versatile *if (1)* $g \geq \mathrm{AND}$, *(2)* g *is flippable, and (3)* g *is random-self-reducible.*

2.2 Two-party example

Consider the function $\mathrm{VER} \colon \mathbb{Z}_4 \times \mathbb{Z}_4 \to \{0,1\}$ defined by

$$\mathrm{VER}(x, y) = 1 \iff x + y \in \{2, 3\}, \quad \text{for all } x, y \in \mathbb{Z}_4,$$

where the arithmetic is that of $\mathbb{Z}_4$; see Figure 2.

Lemma 11. VER *is versatile.*

Proof. The reduction from AND is simply $\mathrm{AND}(x, y) = \mathrm{VER}(x, y)$. Moreover, VER is flippable because $\neg\mathrm{VER}(x, y) = \mathrm{VER}(x + 2, y)$. To see that VER is random-self-reducible, start with (x, y) and compute as follows. First, choose $(\boldsymbol{x}, \boldsymbol{y})$ uniformly at random from the set $\{(x, y), (1 - x, -y)\}$ so that $\boldsymbol{x} + \boldsymbol{y}$ is uniformly distributed either in the set $\{0, 1\}$ if (x, y) was a 0-input, or in the set $\{2, 3\}$ if (x, y) was a 1-input. Finally, choose a random $\boldsymbol{a} \in \mathbb{Z}_4$ and output $(\boldsymbol{x} + \boldsymbol{a}, \boldsymbol{y} - \boldsymbol{a})$. □

It is not hard to show that VER is in fact a minimum-size example of a versatile function: if $g \colon [a] \times [b] \to \{0,1\}$ is

	0	**1**	**2**	**3**
0	0	0	1	1
1	0	1	1	0
2	1	1	0	0
3	1	0	0	1

Figure 2: Function VER.

versatile then $a, b \geq 4$. Indeed, VER is the smallest two-party function for which our proof of Theorem 1 applies. By comparison, the original proof of Theorem 1 [27] uses a certain subfunction HN $\leq$ 3IND whose communication matrix is illustrated in Figure 3. Thus, somewhat interestingly, our proof yields a result that is incomparable to [27] since we have neither VER $\leq$ HN nor HN $\leq$ VER.

Coincidentally, VER makes an appearance in Sherstov's pattern matrix method [47, §12], too. There, the focus is on exploiting the *matrix-analytic* properties of the communication matrix of VER. By contrast, in this work, we celebrate its *self-reducibility* properties.

3. COMMUNICATION LOWER BOUND

In this section we prove the communication lower bound for two parties (Theorem 1) assuming that g is a versatile gadget. The generalisation to multiple parties (Theorem 2) follows essentially by the same argument; see the full version [24] for details.

Our proof builds on a result of Zhang [51] that lower bounds the two-party communication complexity of a composed function $f \circ g^n$ in terms of the block sensitivity of f. We start by outlining Zhang's approach.

3.1 Functions: Zhang's approach

Zhang [51] proved the following theorem by a reduction from the *unique-disjointness* function UDISJ_n. Here

$$\text{UDISJ}_n = \text{OR}_n \circ \text{AND}^n$$

is the usual set-disjointness function together with the promise that if $\text{UDISJ}_n(a, b) = 1$, then there is a unique coordinate $i \in [n]$ such that $a_i = b_i = 1$. The randomised communication complexity of UDISJ_n is well-known to be $\Theta(n)$ [30, 43, 2]. Zhang's proof works for any gadget g with AND, OR $\leq g$.

THEOREM 12 (ZHANG). *There is a two-party gadget $g : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ such that if $f\colon \{0,1\}^n \to Q$ is a function, then $f \circ g^n$ has communication complexity $\Omega(\text{bs}(f))$.*

The proof runs roughly as follows. Fix an input $\alpha \in \{0,1\}^n$ for f that witnesses the block sensitivity $\text{bs}(f, \alpha) = \text{bs}(f)$. Also, let $B_1, \ldots, B_{\text{bs}} \subseteq [n]$ be the sensitive blocks of f at α. Given an input (a, b) to UDISJ_{bs} the goal in the reduction is for the two parties to compute, without communication, an input (x, y) for $f \circ g^n$ such that

(T1) *0-inputs:* If $\text{UDISJ}_{\text{bs}}(a, b) = 0$, then $g^n(x, y) = \alpha$.
(T2) *1-inputs:* If $\text{UDISJ}_{\text{bs}}(a, b) = 1$ with $a_i = b_i = 1$, then $g^n(x, y) = \alpha^{B_i}$.

Clearly, if we had a reduction $(a, b) \mapsto (x, y)$ satisfying (T1–T2), then the output of $\text{UDISJ}_{\text{bs}}(a, b)$ could be recovered from $(f \circ g^n)(x, y)$. Thus, an ϵ-error protocol for $f \circ g^n$ would imply an ϵ-error protocol for UDISJ_{bs} with the same communication cost.

1	0	0	0	1	1
0	1	0	1	0	1
0	0	1	1	1	0

Figure 3: Function HN.

3.2 Search problems: Our approach

We are going to prove Theorem 1 (restated below) in close analogy to the proof template (T1–T2) above. However, as discussed below, noncritical inputs to search problems introduce new technical difficulties.

THEOREM 1 (TWO-PARTY VERSION). *There is a two-party gadget $g\colon \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ such that if $S \subseteq \{0,1\}^n \times Q$ is any search problem, then $S \circ g^n$ has randomised bounded-error communication complexity $\Omega(\text{bs}_{crit}(S))$.*

Setup.

Fix any versatile gadget $g\colon \mathcal{X} \times \mathcal{Y} \to \{0,1\}$. Let Π be a randomised ϵ-error protocol for a composed search problem $S \circ g^n$. Recall that an input (x, y) for the problem $S \circ g^n$ is *critical* if there is exactly one solution q with $((x, y), q) \in S \circ g^n$. In particular, if $g^n(x, y)$ is critical for S, then (x, y) is critical for $S \circ g^n$. The behaviour of the protocol Π on a critical input (x, y) is predictable: the protocol's output $\Pi(x, y)$ is the unique solution with probability at least $1 - \epsilon$.

However, noncritical inputs (x, y) are much trickier: not only can the distribution of the output $\Pi(x, y)$ be complex, but the distributions of $\Pi(x, y)$ and $\Pi(x', y')$ can differ even if (x, y) and (x', y') encode the same input $g^n(x, y) = g^n(x', y')$ of S. The latter difficulty is the main technical challenge, and we address it by using random-self-reducible gadgets.

Defining a function $f \subseteq S$.

We start by following very closely the initial analysis in the proof of Huynh and Nordström [27]. First, we record for each $\alpha \in \{0,1\}^n$ the *most likely feasible output* of Π on inputs (x, y) that encode α. More formally, for each α we define μ_α to be the uniform distribution on the set of preimages of α, i.e.,

$$\mu_\alpha \text{ is uniform on } \{(x, y) : g^n(x, y) = \alpha\}. \tag{1}$$

Alternatively, this can be viewed as a product distribution

$$\mu_\alpha = \mu_{\alpha_1} \times \mu_{\alpha_2} \times \cdots \times \mu_{\alpha_n}, \tag{2}$$

where μ_z, $z \in \{0,1\}$, is the uniform distribution on $g^{-1}(z)$.

The most likely feasible solution output by Π on inputs $(\boldsymbol{x}, \boldsymbol{y}) \sim \mu_\alpha$ is now captured by a total function $f \subseteq S$ defined by

$$f(\alpha) \;:=\; \operatorname*{arg\,max}_{q:(\alpha,q)\in S} \Pr_{(\boldsymbol{x},\boldsymbol{y})\sim\mu_\alpha} [\,\Pi(\boldsymbol{x}, \boldsymbol{y}) = q\,]. \tag{3}$$

Here, ties are broken arbitrarily and the randomness is taken over both $(\boldsymbol{x}, \boldsymbol{y}) \sim \mu_\alpha$ and the random coins of the protocol

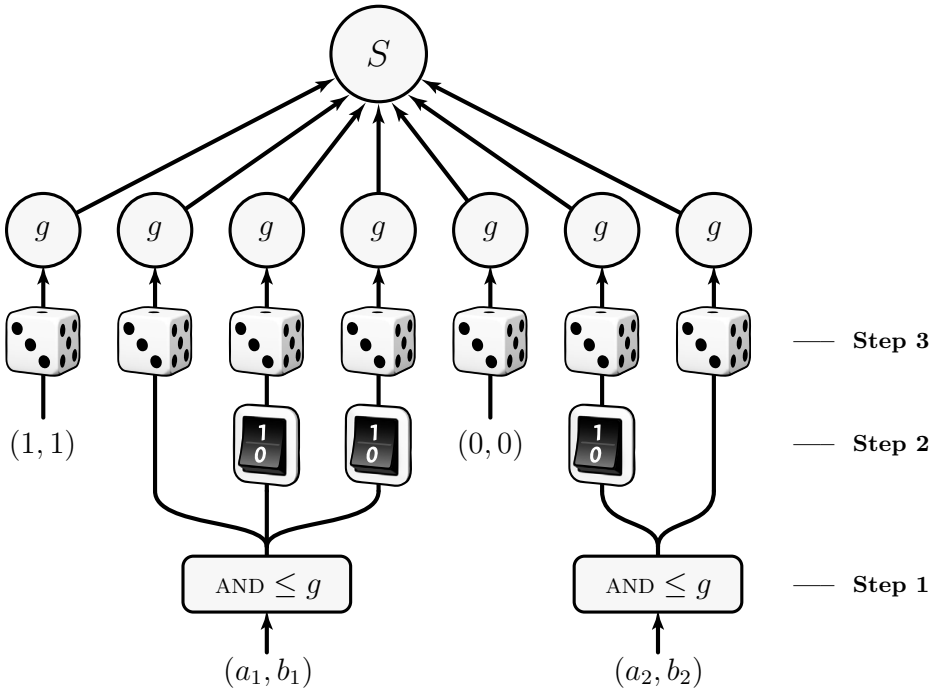

Figure 4: The reduction $(a, b) \mapsto (x, y)$. In this example bs $= 2$ and $n = 7$. The critical input is $\alpha = 1011010$ and the two sensitive blocks are $B_1 = \{2, 3, 4\}$ and $B_2 = \{6, 7\}$. The input pair (a_i, b_i), $i = 1, 2$, is plugged in for the block B_i.

Π. (Note that, in general, the most likely output of $\Pi(\boldsymbol{x}, \boldsymbol{y})$ may not be feasible. However, above, we explicitly pick out the most likely *feasible* solution. Thus, f is indeed a subfunction of S.)

The sensitive critical input.

We can now use the critical block sensitivity of S: there is a critical input α such that $\mathrm{bs}(f, \alpha) \geq \mathrm{bs}_{crit}(S)$. Let $B_1, \ldots, B_{\mathrm{bs}} \subseteq [n]$ be the sensitive blocks with $f(\alpha^{B_i}) \neq f(\alpha)$.

LEMMA 13. *The protocol Π can distinguish between μ_α and $\mu_{\alpha^{B_i}}$ in the sense that*

$$(\boldsymbol{x}, \boldsymbol{y}) \sim \mu_\alpha \quad \Longrightarrow \quad \Pr[\,\Pi(\boldsymbol{x}, \boldsymbol{y}) = f(\alpha)\,] \geq 1 - \epsilon, \quad (4)$$

$$(\boldsymbol{x}, \boldsymbol{y}) \sim \mu_{\alpha^{B_i}} \quad \Longrightarrow \quad \Pr[\,\Pi(\boldsymbol{x}, \boldsymbol{y}) = f(\alpha)\,] \leq 1/2. \quad (5)$$

PROOF. The consequent in the first property (4) is true even for each individual (x, y) in the support of μ_α since α is critical. To see that the second property (5) is true, suppose for a contradiction that we had $\Pr[\,\Pi(\boldsymbol{x}, \boldsymbol{y}) = f(\alpha)\,] > 1/2$ for $(\boldsymbol{x}, \boldsymbol{y}) \sim \mu_{\alpha^{B_i}}$. By averaging, there is a fixed input (x, y) in the support of $\mu_{\alpha^{B_i}}$ such that $\Pr[\,\Pi(x, y) = f(\alpha)\,] > 1/2$. By the correctness of Π (i.e., $1 - \epsilon > 1/2$) this implies that $f(\alpha)$ is feasible for α^{B_i}. Thus, $f(\alpha)$ is the most likely feasible solution output by $\Pi(\boldsymbol{x}, \boldsymbol{y})$, that is, $f(\alpha^{B_i}) = f(\alpha)$ by the definition (3). But this contradicts the fact that f is sensitive to B_i at α. □

The reduction.

Lemma 13 suggests a reduction strategy analogous to the template (T1–T2) of Section 3.1. Given an input (a, b) for $\mathrm{UDISJ}_{\mathrm{bs}}$ our goal is to describe a randomised reduction $(a, b) \mapsto (\boldsymbol{x}, \boldsymbol{y})$ such that

(P1) *0-inputs:* If $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 0$, then $(\boldsymbol{x}, \boldsymbol{y}) \sim \mu_\alpha$.
(P2) *1-inputs:* If $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 1$ with $a_i = b_i = 1$, then $(\boldsymbol{x}, \boldsymbol{y}) \sim \mu_{\alpha^{B_i}}$.

Suppose for a moment that we had a reduction with properties (P1–P2). Let Π' be the protocol that on input (a, b) first applies the reduction $(a, b) \mapsto (\boldsymbol{x}, \boldsymbol{y})$ with properties (P1–P2), then runs Π on $(\boldsymbol{x}, \boldsymbol{y})$, and finally outputs 0 if $\Pi(\boldsymbol{x}, \boldsymbol{y}) = f(\alpha)$ and 1 otherwise. Lemma 13 tells us that

- If $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 0$, then $\Pi'(a, b) = 0$ with probability at least $1 - \epsilon$.
- If $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 1$, then $\Pi'(a, b) = 1$ with probability at least $1/2$.

The error probability of Π' can be bounded away from $1/2$ by repeating Π' twice and outputting 0 iff both runs of Π' output 0. (Here we are assuming that ϵ is small enough, say at most $1/4$. If not, we can use some other standard success probability boosting tricks.) This gives a randomised protocol for $\mathrm{UDISJ}_{\mathrm{bs}}$ with the same communication cost (up to constants) as that of Π. Theorem 1 follows.

Indeed, it remains to implement a reduction $(a, b) \mapsto (\boldsymbol{x}, \boldsymbol{y})$ satisfying (P1–P2). We do it in three steps; see Figure 4.

Step 1: On input $(a, b) = (a_1 \ldots a_{\mathrm{bs}}, b_1 \ldots b_{\mathrm{bs}})$ to $\mathrm{UDISJ}_{\mathrm{bs}}$ we first take each pair (a_i, b_i) through the reduction $\mathrm{AND} \leq g$ to obtain instances $(a'_1, b'_1), \ldots, (a'_{\mathrm{bs}}, b'_{\mathrm{bs}})$ of g. Note that

- if $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 0$, then $g(a'_i, b'_i) = 0$ for all i;
- if $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 1$, then there is a unique i with $g(a'_i, b'_i) = 1$.

Step 2: Next, the instances (a'_i, b'_i) are used to populate a vector $(x, y) = (x_1 \dots x_n, y_1 \dots y_n)$ carrying n instances of g, as follows. The instance (a'_i, b'_i) is plugged in for the coordinates $j \in B_i$ with the copies corresponding to $\alpha_j = 1$ flipped. That is, we define for $j \in B_i$:

- if $\alpha_j = 0$, then $(x_j, y_j) := (a'_i, b'_i)$;
- if $\alpha_j = 1$, then $(x_j, y_j) := (\pi_A(a'_i), \pi_B(b'_i))$, where (π_A, π_B) is the reduction $\neg g \leq g$.

For $j \notin \cup_i B_i$ we simply fix an arbitrary $(x_j, y_j) \in g^{-1}(\alpha_j)$. We now have that

- if $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 0$, then $g^n(x, y) = \alpha$;
- if $\mathrm{UDISJ}_{\mathrm{bs}}(a, b) = 1$ with $a_i = b_i = 1$, then $g^n(x, y) = \alpha^{B_i}$.

Step 3: Finally, we apply a random-self-reduction independently for each component (x_i, y_i) of (x, y): this maps a z-input (x_i, y_i) to a uniformly random z-input $(\boldsymbol{x}_i, \boldsymbol{y}_i) \sim \mu_z$. The result is a random vector $(\boldsymbol{x}, \boldsymbol{y})$ that has a distribution of the form (2) and matches our requirements (P1–P2), as desired. This concludes the proof of Theorem 1.

Acknowledgements

We thank Yuval Filmus for ideas on versatile multiparty gadgets, and Jakob Nordström and Thomas Watson for providing helpful suggestions based on an early draft of this work. We also thank Anil Ada, Paul Beame, Trinh Huynh, and Robert Robere for discussions, and finally the STOC reviewers for useful comments.

This research was supported in part by NSERC. The first author also acknowledges support from Alfred B. Lehman Graduate Scholarship.

4. REFERENCES

[1] L. Babai, P. Frankl, and J. Simon. Complexity classes in communication complexity theory. In *Proceedings of the 27th Symposium on Foundations of Computer Science (FOCS)*, pages 337–347. IEEE, 1986. doi:10.1109/SFCS.1986.15.

[2] Z. Bar-Yossef, T. Jayram, R. Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *Journal of Computer and System Sciences*, 68(4):702–732, 2004. doi:10.1016/j.jcss.2003.11.006.

[3] P. Beame, C. Beck, and R. Impagliazzo. Time-space tradeoffs in resolution: Superpolynomial lower bounds for superlinear space. In *Proceedings of the 44th Symposium on Theory of Computing (STOC)*, pages 213–232, New York, NY, USA, 2012. ACM. doi:10.1145/2213977.2213999.

[4] P. Beame, T. Huynh, and T. Pitassi. Hardness amplification in proof complexity. In *Proceedings of the 42nd Symposium on Theory of Computing (STOC)*, pages 87–96. ACM, 2010. doi:10.1145/1806689.1806703.

[5] P. Beame, T. Pitassi, and N. Segerlind. Lower bounds for Lovász–Schrijver systems and beyond follow from multiparty communication complexity. *SIAM Journal on Computing*, 37(3):845–869, 2007. doi:10.1137/060654645.

[6] C. Beck, J. Nordström, and B. Tang. Some trade-off results for polynomial calculus (extended abstract). In *Proceedings of the 45th Symposium on Theory of Computing (STOC)*, pages 813–822. ACM, 2013. doi:10.1145/2488608.2488711.

[7] E. Ben-Sasson and J. Nordström. Understanding space in proof complexity: Separations and trade-offs via substitutions (extended abstract). In *Proceedings of the 2nd Symposium on Innovations in Computer Science (ICS)*, pages 401–416. Tsinghua University Press, 2011.

[8] E. Ben-Sasson and A. Wigderson. Short proofs are narrow—resolution made simple. *Journal of the ACM*, 48(2):149–169, 2001. doi:10.1145/375827.375835.

[9] M. Braverman, F. Ellen, R. Oshman, T. Pitassi, and V. Vaikuntanathan. A tight bound for set disjointness in the message-passing model. In *Proceedings of the 54th Symposium on Foundations of Computer Science (FOCS)*, pages 668–677. ACM, 2013. doi:10.1109/FOCS.2013.77.

[10] H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288(1):21–43, 2002. doi:10.1016/S0304-3975(01)00144-X.

[11] A. Chakrabarti, Y. Shi, A. Wirth, and A. C.-C. Yao. Informational complexity and the direct sum problem for simultaneous message complexity. In *Proceedings of the 42nd Symposium on Foundations of Computer Science (FOCS)*, pages 270–278. IEEE, 2001. doi:10.1109/SFCS.2001.959901.

[12] S. M. Chan and A. Potechin. Tight bounds for monotone switching networks via Fourier analysis. In *Proceedings of the 44th Symposium on Theory of Computing (STOC)*, pages 495–504. ACM, 2012. doi:10.1145/2213977.2214024.

[13] A. K. Chandra, M. L. Furst, and R. J. Lipton. Multi-party protocols. In *Proceedings of the 15th Symposium on Theory of Computing (STOC)*, pages 94–99. ACM, 1983. doi:10.1145/800061.808737.

[14] M. Charikar, K. Makarychev, and Y. Makarychev. Integrality gaps for Sherali–Adams relaxations. In *Proceedings of the 41st Symposium on Theory of Computing (STOC)*, pages 283–292. ACM, 2009. doi:10.1145/1536414.1536455.

[15] A. Chattopadhyay and T. Pitassi. The story of set disjointness. *SIGACT News*, 41(3):59–85, 2010. doi:10.1145/1855118.1855133.

[16] V. Chvátal. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics*, 4(4):305–337, 1973. doi:10.1016/0012-365X(73)90167-2.

[17] U. Feige, J. Killian, and M. Naor. A minimal model for secure computation. In *Proceedings of the 26th Symposium on Theory of Computing (STOC)*, pages 554–563. ACM, 1994. doi:10.1145/195058.195408.

[18] Y. Filmus, T. Pitassi, R. Robere, and S. A. Cook. Average case lower bounds for monotone switching networks. In *Proceedings of the 54th Symposium on Foundations of Computer Science (FOCS)*, pages 598–607. ACM, 2013. doi:10.1109/FOCS.2013.70.

[19] A. Frieze. Edge-disjoint paths in expander graphs. *SIAM Journal on Computing*, 30(6):1790–1801, 2001. doi:10.1137/S0097539700366103.

[20] A. Frieze and L. Zhao. Optimal construction of edge-disjoint paths in random regular graphs. *Combinatorics, Probability and Computing*, 9:241–263,

4 2000. doi:10.1017/S0963548300004284.

[21] K. Georgiou, A. Magen, T. Pitassi, and I. Tourlakis. Integrality gaps of $2 - o(1)$ for vertex cover SDPs in the Lovász–Schrijver hierarchy. *SIAM Journal on Computing*, 39(8):3553–3570, 2010. doi:10.1137/080721479.

[22] M. Goldmann and J. Håstad. A simple lower bound for monotone clique using a communication game. *Information Processing Letters*, 41(4):221–226, 1992. doi:10.1016/0020-0190(92)90184-W.

[23] R. E. Gomory. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society*, 64:275–278, 1958.

[24] M. Göös and T. Pitassi. Communication lower bounds via critical block sensitivity. arXiv:1311.2355, 2013.

[25] D. Grigoriev. Linear lower bound on degrees of Positivstellensatz calculus proofs for the parity. *Theoretical Computer Science*, 259(1–2):613–622, 2001. doi:10.1016/S0304-3975(00)00157-2.

[26] A. Gronemeier. Asymptotically optimal lower bounds on the NIH-multi-party information complexity of the AND-function and disjointness. In *Proceedings of the 26th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 505–516, 2009. doi:10.4230/LIPIcs.STACS.2009.1846.

[27] T. Huynh and J. 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 (STOC)*, pages 233–248. ACM, 2012. doi:10.1145/2213977.2214000.

[28] R. Impagliazzo, T. Pitassi, and A. Urquhart. Upper and lower bounds for tree-like cutting planes proofs. In *Proceedings of the 9th Symposium on Logic in Computer Science (LICS)*, pages 220–228. IEEE, 1994. doi:10.1109/LICS.1994.316069.

[29] S. Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27 of *Algorithms and Combinatorics*. Springer, 2012.

[30] B. Kalyanasundaram and G. Schnitger. The probabilistic communication complexity of set intersection. *SIAM Journal on Discrete Mathematics*, 5(4):545–557, 1992. doi:10.1137/0405044.

[31] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require super-logarithmic depth. In *Proceedings of the 20th Symposium on Theory of Computing (STOC)*, pages 539–550. ACM, 1988. doi:10.1145/62212.62265.

[32] E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, 1997.

[33] J. B. Lasserre. An explicit SDP relaxation for nonlinear 0-1 programs. In *Proceedings of the 8th International Conference on Integer Programming and Combinatorial Optimization (IPCO)*, volume 2081 of *Lecture Notes in Computer Science*, pages 293–303. Springer, 2001. doi:10.1007/3-540-45535-3_23.

[34] L. Lovász, M. Naor, I. Newman, and A. Wigderson. Search problems in the decision tree model. *SIAM Journal on Discrete Mathematics*, 8(1):119–132, 1995. doi:10.1137/S0895480192233867.

[35] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0–1 optimization. *SIAM Journal on Optimization*, 1(2):166–190, 1991. doi:10.1137/0801013.

[36] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988. doi:10.1007/BF02126799.

[37] J. Nordström. Pebble games, proof complexity, and time-space trade-offs. *Logical Methods in Computer Science*, 9(3:15):1–63, 2013. doi:10.2168/LMCS-9(3:15)2013.

[38] N. Pippenger. Communication networks. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume A, chapter 15, pages 805–833. Elsevier, 1990.

[39] A. Potechin. Bounds on monotone switching networks for directed connectivity. In *Proceedings of the 51st Symposium on Foundations of Computer Science (FOCS)*, pages 553–562. IEEE, 2010. doi:10.1109/FOCS.2010.58.

[40] R. Raz and P. McKenzie. Separation of the monotone NC hierarchy. *Combinatorica*, 19(3):403–435, 1999. doi:10.1007/s004930050062.

[41] R. Raz and A. Wigderson. Probabilistic communication complexity of boolean relations. In *Proceedings of the 30th Symposium on Foundations of Computer Science (FOCS)*, pages 562–567. IEEE, 1989. doi:10.1109/SFCS.1989.63535.

[42] R. Raz and A. Wigderson. Monotone circuits for matching require linear depth. *Journal of the ACM*, 39(3):736–744, 1992. doi:10.1145/146637.146684.

[43] A. A. Razborov. On the distributional complexity of disjointness. *Theoretical Computer Science*, 106(2):385–390, 1992. doi:10.1016/0304-3975(92)90260-M.

[44] G. Schoenebeck. Linear level Lasserre lower bounds for certain k-CSPs. In *Proceedings of the 49th Symposium on Foundations of Computer Science (FOCS)*, pages 593–602. IEEE, 2008. doi:10.1109/FOCS.2008.74.

[45] G. Schoenebeck, L. Trevisan, and M. Tulsiani. A linear round lower bound for Lovász–Schrijver SDP relaxations of vertex cover. In *Proceedings of the 22nd Conference on Computational Complexity (CCC)*, pages 205–216. IEEE, 2007. doi:10.1109/CCC.2007.2.

[46] H. D. Sherali and W. P. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM Journal on Discrete Mathematics*, 3(3):411–430, 1990. doi:10.1137/0403036.

[47] A. A. Sherstov. The pattern matrix method. *SIAM Journal on Computing*, 40(6):1969–2000, 2011. doi:10.1137/080733644.

[48] A. A. Sherstov. Communication lower bounds using directional derivatives. In *Proceedings of the 45th Symposium on Theory of Computing (STOC)*, pages 921–930. ACM, 2013. doi:10.1145/2488608.2488725.

[49] M. Tulsiani. CSP gaps and reductions in the Lasserre hierarchy. In *Proceedings of the 41st Symposium on Theory of Computing (STOC)*, pages 303–312. ACM, 2009. doi:10.1145/1536414.1536457.

[50] A. Urquhart. Hard examples for resolution. *Journal of the ACM*, 34(1):209–219, 1987. doi:10.1145/7531.8928.

[51] S. Zhang. On the tightness of the Buhrman–Cleve–Wigderson simulation. In *Proceedings of the 20th International Symposium on Algorithms and Computation (ISAAC)*, volume 5878 of *Lecture Notes in Computer Science*, pages 434–440. Springer, 2009. doi:10.1007/978-3-642-10631-6_45.