Electronic Colloquium on Computational Complexity, Report No. 165 (2017)

Lifting Nullstellensatz to Monotone Span Programs over any Field

Toniann Pitassi
University of Toronto and IAS
toni@cs.toronto.edu

Robert Robere*
University of Toronto
robere@cs.toronto.edu

November 2, 2017

Abstract

We characterize the size of monotone span programs computing certain "structured" boolean functions by the Nullstellensatz degree of a related unsatisfiable Boolean formula. This yields the first exponential lower bounds for monotone span programs over arbitrary fields, the first exponential separations between monotone span programs over fields of different characteristic, and the first exponential separation between monotone span programs over arbitrary fields and monotone circuits. We also show tight quasipolynomial lower bounds on monotone span programs computing directed st-connectivity over arbitrary fields, separating monotone span programs from non-deterministic logspace and also separating monotone and non-monotone span programs over $GF(2)$. Our results yield the same lower bounds for linear secret sharing schemes due to a known relationship between monotone span programs and linear secret sharing developed by Karchmer and Wigderson [32] and Beimel [7]. To prove our characterization we introduce a new and general tool for lifting *polynomial degree* to *rank* over arbitrary fields, generalizing a result of Sherstov [43].

1 Introduction

Span programs (and monotone span programs) are an elegant model of computation introduced by Karchmer and Wigderson [32] that capture the computational power of linear algebra over a field. To be precise, a span program over a field $\mathbf{F}$ is defined by a matrix M over $\mathbf{F}$ whose rows are labelled with literals over boolean variables $z_1, z_2, \ldots, z_n$ (possibly with repeats); the program is *monotone* if there are no negative literals (i.e. of the form $\neg z_i$) labelling any rows. Given an assignment $z \in \{0,1\}^n$ to these literals, the span program accepts z if the rows of M labelled with literals that are consistent with z span the all-1s vector — with this definition a span program computes a boolean function in the natural way. In the *monotone* case, we have a very interesting model of computation, since monotone span programs use non-monotone operations (algebra over $\mathbf{F}$) to compute monotone functions (recall that a boolean function $f : \{0,1\}^n \to \{0,1\}$ is *monotone* if $x \leq y$ implies $f(x) \leq f(y)$). This makes them surprisingly powerful — for instance, it is known that there are monotone functions f computable by polynomial-size monotone span programs over $GF(2)$, but require super-polynomial size monotone circuits [4]. Further, monotone span programs have an interesting connection to cryptography since they exactly characterize the amount of information that must be shared in *linear secret sharing schemes* [8, 32].

These reasons make monotone span programs an interesting model to study, but due to their power proving strong lower bounds against them is a difficult task. For monotone span programs over the reals, exponential lower bounds were recently proven by [36,42]. However, the strongest lower bounds known for *arbitrary* fields was $n^{\Omega(\log n)}$, shown by Gál [22].

*Research supported by NSERC.

ISSN 1433-8092

Our Contribution. The main contribution of the present work is a new *characterization* of monotone span program size over arbitrary fields for certain "structured" boolean functions. This characterization allows us to resolve a number of open problems about the complexity of monotone span programs (and, therefore, linear secret sharing schemes), and generalize the main results of [36, 42] to arbitrary fields. To summarize:

1. We show, for every field $\mathbf{F}$, that the $\mathbf{F}$-monotone span program size of the *directed st-connectivity* function STCONN is $n^{\Theta(\log n)}$. This is notable as polynomial-size *non-monotone* span programs over $GF(2)$ are known to be able to compute STCONN [44], and thus we give the first field-independent superpolynomial separation between monotone span programs and non-monotone span programs — further, this shows that monotone span programs over any field can be weaker than monotone non-deterministic logspace. Previously this lower bound for STCONN was only known for real span programs [42], and it was not known whether or not there existed a field $\mathbf{F}$ such that monotone span programs over $\mathbf{F}$ could efficiently simulate monotone polynomial-size circuits.

2. For every field $\mathbf{F}$ we show that the $\mathbf{F}$-monotone span program size of the GEN function is $n^{\Theta(n^{\varepsilon})}$ for some fixed constant $\varepsilon > 0$. Since GEN is computable by polynomial-size monotone circuits this yields the first superpolynomial separation between $\mathbf{F}$-monotone span program size and monotone circuit size for all $\mathbf{F}$; once again, this was previously only known for real span programs [42].

3. Finally, for each field $\mathbf{F}$ of finite characteristic, we construct an explicit monotone function f computable in NP such that $\mathsf{mSP}_{\mathbf{F}}(f) = O(\mathsf{poly}(n))$, but for every field $\mathbf{F}'$ of characteristic different from $\mathbf{F}$ we have $\mathsf{mSP}_{\mathbf{F}'}(f) = 2^{\Omega(n)}$, where $\mathsf{mSP}_{\mathbf{F}}(f)$ denotes the monotone span program size of f over $\mathbf{F}$. This is the first exponential separation between monotone span programs of different characteristic (indeed, the lower bound is in fact *strongly* exponential in the sense of [36], and so is tight up to constants in the exponent for *any* monotone function). The best previous separations between monotone span programs of different characteristic are due to Beimel and Weinreb [10], who exhibited a similar separation result, but in which the lower bound was on the order of $\mathsf{mSP}_{\mathbf{F}'}(f) = n^{\Omega(\sqrt{\log n})}$.

Since these results are easy corollaries of our main theorem we leave their proofs to Appendix B. Further, as our results generalize the results of [42], we obtain as corollaries the lower bounds for monotone switching networks obtained by Potechin and Chan-Potechin [15, 37], the depth hierarchy theorem for monotone NC obtained by Raz and McKenzie [39], and the monotone depth lower bounds for st-connectivity obtained by Karchmer and Wigderson [31].

Let us now discuss some of the ingredients of our characterization. At the core of our results is a new *lifting theorem*. Our lifting theorem is in the style of Raz and McKenzie [39] who showed how to construct from any unsatisfiable CNF $\mathcal{C} = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ over n variables and any "two-party" gadget $g : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ a monotone boolean function $f_{\mathcal{C},g}$ which we will call a *lifted function*. Raz and McKenzie showed that for a particular choice of gadget g, the *monotone circuit depth* of the function $f_{\mathcal{C},g}$ is characterized by the *decision tree depth* of the search problem $\mathsf{Search}(\mathcal{C})$ associated with $\mathcal{C}$ (i.e. given an assignment to the variables of $\mathcal{C}$, output a falsified clause of $\mathcal{C}$) — hence, they *lifted* lower bounds from a "simple" computational model (decision trees) to a "complicated" computational model (monotone boolean circuits). (See the Related Works section for other hardness escalation theorems.)

In the present work we prove such a lifting theorem for monotone span programs. In particular, for any field $\mathbf{F}$, we show that for any unsatisfiable CNF $\mathcal{C}$ and for any "good" gadget g the minimum degree of any Nullstellenstaz refutation of $\mathcal{C}$ over $\mathbf{F}$ characterizes the size of the smallest monotone span program computing $f_{\mathcal{C},g}$ over $\mathbf{F}$. The open problems above are then resolved by appealing to the broad literature of Nullstellensatz lower bounds [6, 12, 13, 18, 41].

To be more precise, let $\mathcal{P} = \{p_1 = 0, p_2 = 0, \ldots, p_m = 0\}$ be an unsatisfiable system of polynomial equations over $\mathbf{F}[z_1, z_2, \ldots, z_n]$. A *Nullstellensatz refutation* of $\mathcal{P}$ is given by a set of polynomials

$q_1, q_2, \ldots, q_m$ such that

$$\sum_{i=1}^{m} p_i q_i = 1. \tag{1}$$

The *degree* of the refutation is $\max_i \deg(p_i q_i)$, and the Nullstellensatz degree of $\mathcal{P}$ is the minimum degree $\mathsf{NS}_{\mathbf{F}}(\mathcal{P})$ of any refutation of $\mathcal{P}$. For an unsatisfiable CNF $\mathcal{C}$, we let $\mathsf{NS}_{\mathbf{F}}(\mathcal{C})$ denote the minimum degree of any Nullstellensatz refutation of $\mathcal{C}$ encoded as a system of polynomial equations.

To lift the Nullstellensatz degree, we use an interesting characterization of monotone span program size given by Gál [22]. Let $f : \{0,1\}^n \to \{0,1\}$ be a monotone boolean function, and for any $i \in [n]$ define

$$X_i = \left\{(x, y) \in f^{-1}(1) \times f^{-1}(0) \mid x_i = 1, y_i = 0\right\}$$

to be the *coordinate rectangle* of the ith input. An *algebraic tiling* of f is given by a sequence of n matrices $A_1, A_2, \ldots, A_n$ over $\mathbf{F}$, each of size $|f^{-1}(1)| \times |f^{-1}(0)|$, such that all non-zero entries of A_i are indexed by X_i and

$$\sum_{i=1}^{n} A_i = \mathbf{1} \tag{2}$$

where $\mathbf{1}$ is the all-1s matrix. The *size* of an algebraic tiling is $\sum_{i=1}^{n} \mathrm{rank}_{\mathbf{F}}(A_i)$, and the *algebraic tiling number* of f is the minimum size $\chi_{\mathbf{F}}(f)$ of any algebraic tiling of f. Gál showed that the algebraic tiling number of f is exactly the size of the smallest monotone[1] span program computing f. Superficially, one might expect that there is a connection between algebraic tiling and Nullstellensatz given the similiarities of the expressions (1) and (2).

Using this measure, Gál was able to show that a simple rank-based complexity measure of Razborov [40] lower-bounded monotone span program size, and this measure also plays an important role in the proof of our main theorem. Let $\mathbf{F}$ be any field, and let A be any $|f^{-1}(1)| \times |f^{-1}(0)|$ matrix over $\mathbf{F}$. The *rank measure* of f at A is defined to be

$$\mu_{\mathbf{F}}(f, A) := \frac{\mathrm{rank}_{\mathbf{F}}(A)}{\max\limits_{i \in [n]} \mathrm{rank}_{\mathbf{F}}(A \restriction X_i)}$$

where $A \restriction X_i$ denotes the submatrix of A obtained by zeroing all entries of A outside of X_i. Let $\mu_{\mathbf{F}}(f) := \max_A \mu_{\mathbf{F}}(f, A)$ denote the maximal[2] rank measure of f over all matrices A.

Barring the definition of a "good" gadget (see Section 3), we are now ready to state our characterization. We note that there are small and simple gadgets that are good, and when we refer to the rank of a gadget $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ we mean the rank of g when treated as an $|\mathcal{X}| \times |\mathcal{Y}|$ matrix over $\mathbf{F}$.

Theorem 1.1. *Let $\mathcal{C}$ be a constant-width unsatisfiable CNF on n variables, and let $\mathbf{F}$ be any field. For any good gadget g over $\mathbf{F}$ with rank$(g) = n^2$, the lifted function $f_{\mathcal{C},g}$ satisfies*

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = n^{\Theta(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))}.$$

Further, if $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \Theta(n)$, then for any good gadget g (of sufficiently large but constant rank),

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = 2^{\Theta(n)}.$$

[1] Gál also gave a similar characterization of non-monotone span program size.

[2] Note that the rank of A is integral and bounded by $\min\left\{|f^{-1}(1)|, |f^{-1}(0)|\right\}$ and thus we can safely place a maximum instead of a supremum.

In the process of proving our main theorem, we prove an interesting technical result that we hope will have other applications. If $p \in \mathbf{F}[z_1, z_2, \ldots, z_n]$ is a polynomial over $\mathbf{F}$ and $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ is a gadget then we can create a matrix $p \circ g^n : \mathcal{X}^n \times \mathcal{Y}^n \to \mathbf{F}$ (called a *pattern matrix*, following Sherstov [43]) in the natural way by composing g with p:

$$p \circ g^n := [p(g(x_1, y_1), g(x_2, y_2), \ldots g(x_n, y_n))]_{x,y \in \mathcal{X}^n \times \mathcal{Y}^n}.$$

We show that when g is good then the rank of the matrix $p \circ g^n$ can be calculated directly from the set of monomials occurring in p.

Theorem 1.2. *Let $p \in \mathbf{F}[z_1, z_2, \ldots, z_n]$ be a multilinear polynomial and let $\mathbf{F}$ be a field. For any good gadget $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ we have*

$$\mathit{rank}_{\mathbf{F}}(p \circ g^n) = \sum_{S : \hat{p}(S) \neq 0} \mathit{rank}_{\mathbf{F}}(g)^{|S|}$$

where $\hat{p}(S)$ denotes the coefficient of the monomial $\prod_{i \in S} z_i$ in p.

A special case of this theorem follows from a result of Sherstov [43, Theorem 4.3], however, his result only works for real polynomials p and for a specific choice of gadget g. In contrast, our result works for arbitrary fields and for any gadget g satisfying a general condition (in fact, Sherstov's gadget satisfies our general condition — see Section 4 for details).

Related Work. Span programs were introduced by Karchmer and Wigderson [32], who also showed a connection with secret sharing schemes and produced the first superlinear lower bounds on *non-monotone* span program size. Monotone span programs have a long history of lower bounds. Shortly after Karchmer and Wigderson's paper, Csirmaz [20] proved an $\Omega(n^2/\log n)$ lower bound on monotone span program size. Beimel et al. [9] gave a lower bound of $n^{5/2}$, and then Babai et al. [3] proved the first superpolynomial lower bound on the order of $n^{\Omega(\log n/\log\log n)}$. Each of these results were obtained by direct combinatorial arguments, which were simplified and improved by Gál to $n^{\Omega(\log n)}$ [22]. In the same paper, Gál observed the connection between monotone span programs and the rank measure, and this connection was further investigated by Gál and Pudlák [23]. The superpolynomial lower bounds cited above only applied to functions computable in NP; Beimel and Weinreb [10] gave quasipolynomial lower bounds $n^{\Omega(\sqrt{\log n})}$ for a monotone function in uniform NC^2, establishing that monotone span programs can be weaker than polynomial time. Pudlák and Sgall [38] made the first connection between span programs and Nullstellensatz degree in the context of feasible interpolation. The first exponential lower bounds for monotone span programs were proved by Robere, Pitassi, Rossman and Cook [42], who showed exponential lower bounds for *real* span programs. Later, Pitassi and Robere [36] proved the first *strongly exponential* lower bounds for an explicit monotone function (in NP), again over the reals.

These last results bear further discussion, as they were a direct inspiration for the present paper. The results of [36, 42] show that lower bounds on the rank measure $\mu_{\mathbf{R}}(f_{\mathcal{C},g})$ of the lifted function $f_{\mathcal{C},g}$ can be obtained from query complexity lower bounds for $\mathcal{C}$, using a new query measure called the *algebraic gap complexity* $\mathsf{gap}(\mathcal{C})$. This can be seen as establishing one direction of a lifting theorem from algebraic gaps to the rank measure over the reals. For the purpose of proving lower bounds, it was then required to prove strong lower bounds on the algebraic gap, which was done directly for each application in an ad-hoc manner. Here, we prove that the algebraic gap measure is exactly the same as the well-studied Nullstellensatz degree measure in proof complexity, obtaining a full two-way lifting theorem from Nullstellensatz degree to the rank measure over every field. This allows us to obtain all applications (the new ones mentioned above as well as all of the old ones) by simply plugging in the appropriate gadget and unsatisfiable $\mathcal{C}$, and then applying known Nullstellensatz degree bounds.

Hardness escalation techniques like those employed in the present paper are a rapidly growing area of research in complexity theory, and are often used to study the amount of communication needed to compute *composed problems* of the form $F \circ g^n := F(g(x_1, y_1), g(x_2, y_2), \ldots, g(x_n, y_n))$ where F is an n-input function or relation and g is some two-input "gadget". Such composed problems yield a natural two-player communication task: Alice receives $x \in \mathcal{X}^n$, Bob receives $y \in \mathcal{Y}^n$, and their goal is to evaluate F on the input

$$z = g(x_1, y_1)g(x_2, y_2) \cdots g(x_n, y_n)$$

using a minimal amount of communication about their inputs. For many models of communication, if one chooses the "right" gadget g it is possible to show that the communication complexity of the composed function $F \circ g^n$ is closely related to the query complexity of F in some appropriate query model — a typical query model studied is a *decision tree*, which measures the number of input bits of F that need to be queried before we can determine the output of f. Such a result is often very powerful since query models are usually much easier to study than communication models.

Lifting theorems have introduced powerful new tools into complexity theory, and have recently led to the resolution of open problems in many areas of theoretical computer science and discrete mathematics, including: graph theory [24], linear programming formulations for combinatorial optimization [33, 34], circuit complexity and cryptography [26, 36, 39, 42], proof complexity [21, 26, 29], game theory [5], and communication complexity [1, 17, 27, 28, 43]. Moreover the field has led to a revival of query complexity, with new techniques leading to the resolution of some longstanding open problems [1, 2, 24].

2 Preliminaries

Let $\mathcal{Z}$ be a set, and let n be a positive integer. We will use the standard notation of $\mathcal{Z}^n$ to represent the set of all n-tuples over $\mathcal{Z}$, and the less-standard notation $\mathcal{Z}^{\leq n}$ to denote the set of all tuples of length at most n over $\mathcal{Z}$. If $z \in \mathcal{Z}^n$ then z_i denotes the ith element of the tuple z, and if $A \subseteq [n]$ then z_A is the tuple of elements in z indexed by A.

Let $\mathbf{F}$ be a field. It will be useful to think of matrices over $\mathbf{F}$ as having their rows and columns being indexed by more general objects. Thus, if $\mathcal{X}, \mathcal{Y}$ are sets then we consider functions $A : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ as $|\mathcal{X}| \times |\mathcal{Y}|$ matrices, where the rows of A are indexed by elements of $\mathcal{X}$ and columns of A are indexed by $\mathcal{Y}$. To simplify notation, we will refer to such a function A as a $\mathcal{X} \times \mathcal{Y}$ matrix over $\mathbf{F}$, and use regular function notation (e.g. $A(x, y)$ for $x, y \in \mathcal{X} \times \mathcal{Y}$) to index into such matrices. We let $\mathbf{1}_{\mathcal{X},\mathcal{Y}}$ denote the $\mathcal{X} \times \mathcal{Y}$ all-$1s$ matrix, but will often leave out the subscript if the dimensions of the matrix are clear from the context.

If A is an $m \times n$ matrix and B is a $p \times q$ matrix then the *Kronecker product* $A \otimes B$ is the $mp \times nq$ matrix defined by

$$A \otimes B = [A(i, j) \cdot B]_{i \in [m], j \in [n]}.$$

If we think of A and B as mappings $A : [m] \times [n] \to \mathbf{F}$, $B : [p] \times [q] \to \mathbf{F}$, then the Kronecker product has a natural interpretation as the mapping $A \otimes B : ([m] \times [p]) \times ([n] \times [q]) \to \mathbf{F}$ defined by

$$(A \otimes B)((i, k), (j, \ell)) = A(i, j)B(k, \ell).$$

If $\mathcal{X}, \mathcal{Y}$ are sets then a *combinatorial rectangle* in $\mathcal{X} \times \mathcal{Y}$ is a subset $R \subseteq \mathcal{X} \times \mathcal{Y}$ for which we can write $R = X \times Y$ for some subsets $X \subseteq \mathcal{X}, Y \subseteq \mathcal{Y}$. If $A : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ is a matrix and R is a rectangle in $\mathcal{X} \times \mathcal{Y}$ then we let $A{\restriction}R$ denote the submatrix of A indexed by elements of R. It will be formally convenient to think of $A{\restriction}R$ as having the same dimensions as A, and thus we formally define $A{\restriction}R : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ by

$$(A{\restriction}R)(x, y) = \begin{cases} A(x, y) & \text{if } (x, y) \in R \\ 0 & \text{otherwise.} \end{cases}$$

We say that A is *embedded* in R if $A = A\restriction R$ — that is, all non-zero entries of A are indexed by R.

If $x, y \in \{0,1\}^n$ then we write $x \leq y$ if $x_i \leq y_i$ for all i. Let $f : \{0,1\}^n \to \{0,1\}$ be a boolean function, a *partial* boolean function is a function $f : \{0,1\}^n \to \{0,1,*\}$ (informally, if $f(x) = *$ then we "don't care" what the output of the function is). A total boolean function $f : \{0,1\}^n \to \{0,1\}$ is *monotone* if $f(x) \leq f(y)$ whenever $x \leq y$; a partial boolean function $f : \{0,1\}^n \to \{0,1,*\}$ is monotone if it can be extended to a total monotone boolean function by choosing $\{0,1\}$-assignments for the $*$ outputs.

If $f(x) = 1$ we call x an *accepting instance* or a *yes instance*, while if $f(x) = 0$ then we call x a *rejecting instance* or a *no instance*. If f is monotone and $x \in f^{-1}(1), y \in f^{-1}(0)$ then there exists an index $i \in [n]$ such that $x_i = 1, y_i = 0$, as otherwise we would have $x \leq y$, contradicting the fact that f is monotone.

2.1 Circuit Complexity: Karchmer-Wigderson Games and Monotone Span Programs

In this section we review some definitions from circuit complexity. Let $\mathbf{F}$ be a field. An $\mathbf{F}$-*span program* is a computational device for computing boolean functions defined by a matrix A over $\mathbf{F}$ with its rows labelled by boolean literals over variables $z_1, z_2, \ldots, z_n$. Given a span program A, a row vector A_i of A is *consistent* with an input $z \in \{0,1\}^n$ if the literal labelling A_i is set to 1 under z. The span program A then *accepts* an input assignment $z \in \{0,1\}^n$ if the set of rows consistent with z spans the all-1s vector; with this definition a span program A computes a boolean function $f : \{0,1\}^n \to \{0,1\}$ in the natural way. A span program is *monotone* if all literals labelling rows of A are positive, and note that monotone span programs compute monotone functions since adding row vectors can only increase the span. If f is a partial monotone boolean function then we let $\mathsf{mSP}_{\mathbf{F}}(f)$ denote the minimum size of a $\mathbf{F}$-monotone span program computing f.

A set of extremely useful tools in studying the circuit complexity of boolean functions originate in *communication complexity*. Let $f : \{0,1\}^n \to \{0,1,*\}$ be a partial, monotone boolean function, and let $\mathcal{U} = f^{-1}(1), \mathcal{V} = f^{-1}(0)$. The *monotone Karchmer-Wigderson game of* f is the relation

$$\mathsf{KW}^+(f) = \{(x, y, i) \in \mathcal{U} \times \mathcal{V} \times [n] \mid x_i = 1, y_i = 0\}.$$

We think of this relation as a computation task between two parties, Alice and Bob: Alice receives an input $x \in \mathcal{U}$, Bob receives an input $y \in \mathcal{V}$, and they wish to agree on an index i such that $x_i = 1$ and $y_i = 0$. Indeed, the relation $\mathsf{KW}^+(f)$ was introduced by Karchmer and Wigderson [31], who showed that the minimum number of bits that Alice and Bob need to communicate to compute $\mathsf{KW}^+(f)$ is exactly the minimum depth of any monotone circuit[3] computing f.

Similarly, Gál [22] characterized the size of *span programs* computing f using a different complexity measure of the Karchmer-Wigderson game. For any $i \in [n]$ we refer to the set $X_i = \{x \in \mathcal{U} \mid x_i = 1\} \times \{y \in \mathcal{V} \mid y_i = 0\}$ as the *coordinate rectangle* for the input x_i, and note that X_i is a combinatorial rectangle in $\mathcal{U} \times \mathcal{V}$.

Definition 2.1. Let $f : \{0,1\}^n \to \{0,1,*\}$ be a partial monotone boolean function, and let $\mathcal{U} = f^{-1}(1)$, $\mathcal{V} = f^{-1}(0)$. Let $\mathbf{F}$ be a field. If $A : \mathcal{U} \times \mathcal{V} \to \mathbf{F}$ is a matrix and X_i is a coordinate rectangle of $\mathsf{KW}^+(f)$ then A is *embedded* in X_i if A only takes non-zero values inside X_i, i.e. $A = A\restriction X_i$. An *algebraic tiling* of $\mathsf{KW}^+(f)$ is given by a set of matrices $A_1, A_2, \ldots, A_n$ such that

$$\sum_{i=1}^{n} A_i = \mathbf{1}$$

and A_i is embedded in X_i for each i; the *size* of the tiling is $\sum_{i=1}^{n} \mathrm{rank}_{\mathbf{F}}(A_i)$. The *algebraic tiling number* of $\mathsf{KW}^+(f)$, denoted $\chi_{\mathbf{F}}(f)$, is the minimum size of any algebraic tiling of $\mathsf{KW}^+(f)$.

[3] Karchmer and Wigderson also showed that a similar relation characterized non-monotone circuit depth.

Theorem 2.2 (Theorem 3.4 in [22]). *For any partial monotone boolean function f and any field $\mathbf{F}$, $\mathsf{mSP}_{\mathbf{F}}(f) = \chi_{\mathbf{F}}(f)$.*

Using the algebraic tiling number Gál showed that the following measure (originally introduced by Razborov) is a lower bound on span program size.

Definition 2.3. Let $f : \{0,1\}^n \to \{0,1,*\}$ be a partial, monotone boolean function and let $\mathcal{U} = f^{-1}(1), \mathcal{V} = f^{-1}(0)$. Let $\mathbf{F}$ be any field and let A be any $\mathcal{U} \times \mathcal{V}$ matrix over $\mathbf{F}$. Let X_i denote the coordinate rectangle $X_i = \{u \in \mathcal{U} \mid u_i = 1\} \times \{v \in \mathcal{V} \mid v_i = 0\}$ from the relation $\mathsf{KW}^+(f)$. The *rank measure* of f with respect to A is

$$\mu_{\mathbf{F}}(f, A) := \frac{\operatorname{rank}_{\mathbf{F}}(A)}{\max\limits_{i \in [n]} \operatorname{rank}_{\mathbf{F}}(A \restriction X_i)}.$$

Let $\mu_{\mathbf{F}}(f) = \max_A \mu_{\mathbf{F}}(f, A)$.

Theorem 2.4 (Lemma 3.2 in [22]). *For any partial monotone boolean function f and any field $\mathbf{F}$, $\mu_{\mathbf{F}}(f) \leq \chi_{\mathbf{F}}(f)$.*

2.2 Proof Complexity: Nullstellensatz Proofs and Algebraic Gaps

Next we review some preliminaries from proof complexity, and in particular the Nullstellensatz proof system [6]. Let $p \in \mathbf{F}[z_1, z_2, \ldots, z_n]$ be a polynomial over a field $\mathbf{F}$. The polynomial p is *multilinear* if no individual variable appears in p with degree greater than 1. If p is multilinear, it follows that all terms in p are products of variables $\prod_{i \in S} z_i$ for some $S \subseteq [n]$, and thus it has at most 2^n distinct terms. Given a multilinear polynomial p, we will borrow notation from Fourier analysis and let $\hat{p}(S) \in \mathbf{F}$ denote the coefficient of the monomial $z_S := \prod_{i \in S} z_i$ in p. Furthermore, if $\pi : [n] \to \mathbf{F} \cup \{*\}$ is a partial restriction of the variables of p, then we let $p \restriction \pi$ denote the polynomial over the unrestricted variables of π obtained from p in the natural way.

Definition 2.5. Let $\{p_1 = 0, p_2 = 0, \ldots, p_m = 0\}$ be an unsatisfiable system of polynomial equations over variables $z_1, z_2, \ldots, z_m$. A *Nullstellensatz refutation* of the system is given by polynomials $q_1, q_2, \ldots, q_m$ over the same set of variables satisfying

$$\sum_{i=1}^{m} p_i q_i = 1$$

where the equality is syntactic. The *degree* of the refutation is $\max_i \deg(p_i q_i)$.

It is fruitful to compare this definition with Definition 2.1: Nullstellensatz degree is the analogue of the algebraic tiling number for polynomials.

Let $\mathcal{C} = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ be an unsatisfiable CNF over boolean variables $z_1, z_2, \ldots, z_m$. We will need to convert $\mathcal{C}$ to an equivalent system of polynomial equations; here we give two standard encodings that we will use later. The first encoding treats the variables z_i as having $\{0,1\}$ values, and it can be used when the underlying field is arbitrary. If C is a clause we let C^+ denote the set of variables occurring positively in $\mathcal{C}$ and C^- denote the set of variables occurring negatively in $\mathcal{C}$; with this notation we can write

$$C = \bigvee_{z \in C^+} z^1 \vee \bigvee_{z \in C^-} z^0.$$

From C we can define the polynomial equation

$$\mathcal{E}(C) \equiv \prod_{z \in C^+} (1 - z) \prod_{z \in C^-} z = 0,$$

observing that $\mathcal{E}(C)$ is satisfied (over $0/1$ assignments to z_i) if and only if the corresponding assignment satisfies C. We will abuse notation and let $\mathcal{E}(\mathcal{C}) = \{\mathcal{E}(C) \mid C \in \mathcal{C}\} \cup \left\{z_i^2 - z_i = 0\right\}_{i\in[m]}$.

The second encoding treats the variables z_i as having $\{\pm 1\}$ values (in this encoding, -1 is considered to be "True", and 1 is considered to be "False"), and as such can only be used if the underlying field $\mathbf{F}$ satisfies $\mathrm{char}(\mathbf{F}) \neq 2$. Now each clause C is encoded as

$$\mathcal{E}^*(C) \equiv \prod_{z\in C^+} (1+z) \prod_{z\in C^-} (1-z) = 0.$$

Once again, we abuse notation and let $\mathcal{E}^*(\mathcal{C}) = \{\mathcal{E}^*(C) \mid C \in \mathcal{C}\} \cup \left\{z_i^2 - 1 = 0\right\}$.

It is important to note that the choice of encoding does *not* affect the degree of the resulting refutation as one can pass from one encoding to the other in a degree-preserving way. (To go from $\mathcal{E}(C)$ to $\mathcal{E}^*(C)$, simply replace each variable z_i with $y_i = 1 - 2z_i$; the reverse direction is obtained symmetrically.) Thus we can, without loss of generality, define *the* Nullstellensatz degree $\mathsf{NS}_{\mathbf{F}}(\mathcal{C})$ of an unsatisfiable CNF $\mathcal{C}$ as the minimum degree of any Nullstellensatz refutation of either $\mathcal{E}(\mathcal{C})$ or $\mathcal{E}^*(\mathcal{C})$.

One can also consider these two encodings as encodings of $\mathcal{C}$ in $\mathbf{F}[z_1, z_2, \ldots, z_n]/I$ for different ideals I: the first encoding uses the ideal generated by $I_1 = \left\{z_i^2 - z_i\right\}_{i\in[m]}$, and the second uses the ideal generated by $I_2 = \left\{z_i^2 - 1\right\}_{i\in[m]}$. It is well known that all polynomials in $\mathbf{F}[z_1, z_2, \ldots, z_n]$ are equivalent to a multilinear polynomial modulo either of these ideals.

Unsatisfiable CSPs and Search Problems. Just as we can study the complexity of computing a function $f : \{0,1\}^n \to \{0,1\}$ by studying an associated search problem $\mathsf{KW}^+(f)$, one can study the complexity of refuting an unsatisfiable CNF $\mathcal{C}$ by studying an associated search problem $\mathsf{Search}(\mathcal{C})$; we define this search problem next.

A *constraint satisfaction problem* (CSP) is defined by a collection $z_1, z_2, \ldots, z_n$ of variables over a domain $\mathcal{Z}$ and a collection $\mathcal{C} = \{P_1(Z_1), P_2(Z_2), \ldots, P_m(Z_m)\}$ of predicates over these variables; formally, for all i we have $P_i : \mathcal{Z}^t \to \{0,1\}$ and $P_i(Z_i) = P(z_{i_1}, z_{i_2}, \ldots, z_{i_t})$ for some $t \leq n$ and for distinct indices $i_1, i_2, \ldots, i_t$. We say that $\mathcal{C}$ is a k-CSP if every predicate has arity at most k, and $\mathcal{C}$ is *satisfiable* if there is an assignment $z \in \mathcal{Z}^n$ such that $P_i(z) = 1$ for all $i \in [m]$.

If $P : \mathcal{Z}^t \to \{0,1\}$ is a predicate over $\mathcal{Z}$ then a *certificate* of P is a falsifying assignment of P; furthermore, if $\mathcal{C} = \{P_1(Z_1), P_2(Z_2), \ldots, P_m(Z_m)\}$ is a CSP then a *certificate* of $\mathcal{C}$ is a partial restriction $\pi : [n] \to \mathcal{Z} \cup \{*\}$ encoding a certificate of a constraint P_i. If π is a certificate of $\mathcal{C}$ then we let $\mathsf{vars}(\pi) \subseteq [n]$ denote the set of variables which are assigned to values in $\mathcal{Z}$ by π. A certificate π of $\mathcal{C}$ is *consistent* with an assignment $z \in \mathcal{Z}^n$ if $z_i = \pi(i)$ for all $i \in \mathsf{vars}(\pi)$. Let $\mathsf{Cert}(\mathcal{C})$ denote the set of all certificates of $\mathcal{C}$.

Definition 2.6. Let $\mathcal{C}$ be an unsatisfiable CSP on $\mathcal{Z}$-valued variables $z_1, z_2, \ldots, z_n$. The *search problem* associated with $\mathcal{C}$ is the relation

$$\mathsf{Search}(\mathcal{C}) \subseteq \mathcal{Z}^n \times \mathsf{Cert}(\mathcal{C})$$

which contains all pairs $(z, \pi) \in \mathcal{Z}^n \times \mathsf{Cert}(\mathcal{C})$ such that π is consistent with z.

To illustrate this definition we give two examples.

Example 2.7. Let $\mathcal{C} = \{C_1, C_2, \cdots, C_m\}$ be an unsatisfiable k-CNF over variables $z_1, z_2, \ldots, z_n$. The search problem $\mathsf{Search}(\mathcal{C}) \subseteq \{0,1\}^n \times \mathsf{Cert}(\mathcal{C})$ is defined as follows: given a boolean assignment to the variables z, choose any falsified clause C_i and output the assignment to the variables of C_i. (Equivalently, since clauses have a unique falsifying assignment, one can instead just output the index of the falsified clause i.)

Example 2.8. Let $\mathcal{C} = \{p_1 = 0, p_2 = 0, \ldots, p_m = 0\}$ be an unsatisfiable system of polynomials in a polynomial ring $\mathbf{F}[z_1, z_2, \ldots, z_n]$. The search problem $\mathsf{Search}(\mathcal{C})$ is defined as follows: given an assignment $\mathbf{F}^n$ to the variables of $\mathcal{C}$, choose any falsified polynomial equation $p_i = 0$ and output the assignment to the variables appearing in p_i.

Just as we think of Nullstellensatz as the "polynomial analogue" of the algebraic tiling number, we can introduce a "polynomial analogue" of the rank measure (cf. Definition 2.3) using the definition of the search problem $\mathsf{Search}(\mathcal{C})$ above (in particular, using the set of certificates of $\mathcal{C}$).

Definition 2.9. Let $\mathcal{C} = \{p_1 = 0, p_2 = 0, \ldots, p_m = 0\}$ be a system of unsatisfiable polynomial equations and let $\mathbf{F}$ be a field. The *algebraic gap complexity* of $\mathsf{Search}(\mathcal{C})$ is the largest integer $\mathsf{gap}_{\mathbf{F}}(\mathcal{C})$ for which there exists a multilinear polynomial p over $\mathbf{F}$ such that

$$\deg p = n \quad \text{and} \quad \forall \pi \in \mathsf{Cert}(\mathcal{C}), \deg p \restriction \pi \leq n - \mathsf{gap}_{\mathbf{F}}(\mathcal{C}).$$

The prior definition of algebraic gap complexity [36, 42] was for unsatisfiable CNFs $\mathcal{C}$ and not unsatisfiable systems of equations — we state it more generally as we will be considering multiple encodings of CNFs as polynomial systems of equations $\mathcal{E}^*(\mathcal{C})$ and $\mathcal{E}(\mathcal{C})$ (indeed, the previous definition of algebraic gap complexity is equivalent to $\mathsf{gap}_{\mathbf{R}}(\mathcal{E}^*(\mathcal{C}))$). We note that the main difference between $\mathsf{gap}_{\mathbf{F}}(\mathcal{E}(\mathcal{C}))$ and $\mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C}))$ is in the definition of the *certificates*: the certificates of $\mathcal{E}(\mathcal{C})$ are $\{0,1\}$-valued and the certificates of $\mathcal{E}^*(\mathcal{C})$ are $\{\pm 1\}$ valued.

Since the rank measure $\mu_{\mathbf{F}}(f)$ is a lower bound on $\chi_{\mathbf{F}}(f)$ (Theorem 2.4) it is reasonable to expect that $\mathsf{gap}_{\mathbf{F}}(\mathcal{C})$ is a lower bound on $\mathsf{NS}_{\mathbf{F}}(\mathcal{C})$. When each equation $p_i = 0$ has a unique certificate we can show that this is true — this applies in particular when $\mathcal{C}$ is obtained from an unsatisfiable CNF.

Proposition 2.10. *Let $\mathcal{C} = \{p_1 = 0, p_2 = 0, \ldots, p_m = 0\}$ be an unsatisfiable system of polynomial equations and suppose that each equation $p_i = 0$ has a unique certificate. Then $\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) \leq \mathsf{NS}_{\mathbf{F}}(\mathcal{C})$.*

Proof. Let $q_1, q_2, \ldots, q_m$ be any set of polynomials such that $\sum_{i=1}^m p_i q_i = 1$ and $\max_i \deg(p_i q_i) = \mathsf{NS}_{\mathbf{F}}(\mathcal{C})$. Let r be a polynomial witnessing the algebraic gap of $\mathcal{C}$. Multiplying the Nullstellensatz refutation $\sum_{i=1}^m p_i q_i = 1$ by r we can express $r(z) = \sum_{i=1}^m r p_i q_i(z)$. Observe that $(r p_i q_i)(z) \neq 0$ only if z is consistent with the unique certificate π_i of the constraint $p_i = 0$; it follows that $r(z) = \sum_{i=1}^m r p_i q_i(z) = \sum_{i=1}^m (r \restriction \pi_i) p_i q_i(z)$. Using this observation and noting that $\deg r = n$ we get that

$$n = \deg r \leq \max_{i \in [m]} \deg((r \restriction \pi_i) p_i q_i) \leq \max_{i \in [m]} \deg(r \restriction \pi_i) + \deg(p_i q_i) \leq n - \mathsf{gap}_{\mathbf{F}}(\mathcal{C}) + \max_{i \in [m]} \deg(p_i q_i).$$

Rearranging yields $\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) \leq \max_{i \in [m]} \deg(p_i q_i) = \mathsf{NS}_{\mathbf{F}}(\mathcal{C})$. □

2.3 Connecting Proofs and Circuits

Next we connect the search problem $\mathsf{Search}(\mathcal{C})$ associated with unsatisfiable k-CSPs and monotone Karchmer-Wigderson games $\mathsf{KW}^+(f)$. Special cases of the construction in this section have been implicitly used by several other works in the literature [26, 35, 36, 39, 42]; we give a very general presentation in the hope that it will be useful elsewhere.

Let $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ be arbitrary sets, and let $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$ be a relation. We always assume that relations $\mathcal{R}$ are *minimal* in the sense that there are no pairs of "duplicate elements"; i.e. there are no distinct $a, a' \in \mathcal{A}$ such that for all b, c, $(a, b, c) \in \mathcal{R}$ if and only if $(a', b, c) \in \mathcal{R}$ (and the analogous conditions hold for $\mathcal{B}$ and $\mathcal{C}$). The relation $\mathcal{R}$ is *total* if for all $(a, b) \in \mathcal{A} \times \mathcal{B}$ there is a $c \in \mathcal{C}$ such that $(a, b, c) \in \mathcal{R}$, and $\mathcal{R}$ is *rectangular* if for each $c \in \mathcal{C}$ the set

$$R_c = \{(a, b) \in \mathcal{A} \times \mathcal{B} \mid (a, b, c) \in \mathcal{R}\}$$

is either empty or is a combinatorial rectangle.

Now, let $f : \{0,1\}^n \to \{0,1,*\}$ be a partial monotone boolean function and consider the monotone Karchmer-Wigderson game $\mathsf{KW}^+(f)$. It is easy to see that $\mathsf{KW}^+(f)$ is both total and rectangular. In fact, it is not hard to see that these two properties *characterize* monotone Karchmer-Wigderson games in the following sense.

Definition 2.11 (Folklore, [10, 22, 40])**.** Let $\mathcal{X}, \mathcal{Y}$ be sets, and let $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{Y} \times [m]$ be a relation that is total and rectangular. For each $i \in [m]$ let $R_i = X_i \times Y_i \subseteq \mathcal{X} \times \mathcal{Y}$ denote the rectangle corresponding to the output i of the relation, and define a partial monotone boolean function $f_{\mathcal{R}} : \{0,1\}^m \to \{0,1,*\}$ as follows. For each $a \in \mathcal{X}$ define the string $u^a \in \{0,1\}^m$ by setting $u^a_i = 1$ if $a \in X_i$ and $u^a_i = 0$ otherwise. Similarly, for each $b \in \mathcal{Y}$ define the string $v^b \in \{0,1\}^m$ by setting $v^b_i = 0$ if $b \in X_i$ and $v^b_i = 1$ otherwise. Then define

$$f_{\mathcal{R}}(x) = \begin{cases} 1 & \text{if } \exists a \in \mathcal{A} : x = u^a \\ 0 & \text{if } \exists b \in \mathcal{B} : x = v^b \\ * & \text{otherwise.} \end{cases}$$

Proposition 2.12. *Let $\mathcal{X}, \mathcal{Y}$ be sets, and let $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{Y} \times [m]$ be a relation that is total and rectangular. Then $f_{\mathcal{R}}$ is well defined and $\mathsf{KW}^+(f_{\mathcal{R}})$ is equivalent (up to renaming elements of the relation) to $\mathcal{R}$.*

Proof. By way of contradiction suppose that $f_{\mathcal{R}}$ is not well defined and let $(a, b) \in \mathcal{A} \times \mathcal{B}$ be a pair of elements chosen so that $u^a = v^b$. By construction, it follows that there does not exist an $i \in [m]$ such that $(a, b, i) \in \mathcal{R}$, contradicting the totality of $\mathcal{R}$. Furthermore, it is clear that $(a, b, i) \in \mathcal{R}$ if and only if $(u^a, v^b, i) \in \mathsf{KW}^+(f)$. □

With this proposition in mind, consider an unsatisfiable CSP $\mathcal{C}$ with variables taking values in a domain $\mathcal{Z}$ and observe that the relation $\mathsf{Search}(\mathcal{C})$ is total. If $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is a matrix (also called a *gadget*) then we can compose (or *lift*) $\mathsf{Search}(\mathcal{C})$ with g to obtain a new total relation

$$\mathsf{Search}(\mathcal{C}, g) \subseteq \mathcal{X}^n \times \mathcal{Y}^n \times \mathsf{Cert}(\mathcal{C})$$

in the natural way: on input $(x, y) \in \mathcal{X}^n \times \mathcal{Y}^n$, find a certificate $\pi \in \mathsf{Cert}(\mathcal{C})$ that is consistent with the string

$$z = g^n(x, y) := g(x_1, y_1)g(x_2, y_2) \cdots g(x_n, y_n).$$

The lifted search problem $\mathsf{Search}(\mathcal{C}, g)$ is total since the search problem $\mathsf{Search}(\mathcal{C})$ is, however, we cannot immediately apply Proposition 2.12 to construct a boolean function since the lifted problem is not necessarily rectangular. To avoid this issue, we instead consider a canonical version of the search problem obtained by replacing the outputs of $\mathsf{Search}(\mathcal{C}, g)$ with a rectangle covering (in the language of communication complexity, we are showing that the search problem has bounded *certificate complexity*).

Definition 2.13. Let $\mathcal{C} = \{C_1, C_2, \cdots, C_m\}$ be an unsatisfiable k-CSP on variables $z_1, z_2, \ldots, z_n$ with domain $\mathcal{Z}$, and let $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ be any function. The *canonical search problem* is the relation

$$\mathsf{CanSearch}(\mathcal{C}, g) \subseteq \mathcal{X}^n \times \mathcal{Y}^n \times (\mathsf{Cert}(\mathcal{C}) \times \mathcal{X}^{\leq k})$$

defined by

$$((x, y), (\pi, \alpha)) \in \mathsf{CanSearch}(\mathcal{C}, g) \iff z = g^n(x, y) \text{ is consistent with } \pi \text{ and } x{\restriction}_{\mathsf{vars}(\pi)} = \alpha.$$

The canonical search problem $\mathsf{CanSearch}(\mathcal{C}, g)$ also satisfies the rectangularity property and so it can be used to construct monotone boolean functions via Proposition 2.12. To see this, let $\pi \in \mathsf{Cert}(\mathcal{C})$ and $\alpha \in \mathcal{X}^{\leq k}$ be chosen so that there exists $((x, y), (\pi, \alpha)) \in \mathsf{CanSearch}(\mathcal{C}, g)$. We claim the set

$$X_{\pi,\alpha} = \{(x, y) \in \mathcal{X}^n \times \mathcal{Y}^n \mid ((x, y), (\pi, \alpha)) \in \mathsf{CanSearch}(\mathcal{C}, g)\}$$

is a combinatorial rectangle. Towards this, let $(x, y), (x', y') \in X_{\pi,\alpha}$ be two pairs in $X_{\pi,\alpha}$; clearly $x\restriction\mathsf{vars}(\pi) = x'\restriction\mathsf{vars}(\pi) = \alpha$, thus it follows immediately that $(x', y), (x, y') \in X_{\pi,\alpha}$.

Now by applying Proposition 2.12 we obtain from $\mathsf{CanSearch}(\mathcal{C}, g)$ a partial monotone boolean function $f_{\mathsf{CanSearch}(\mathcal{C},g)}$ on $N \leq |\mathsf{Cert}(\mathcal{C})||\mathcal{X}|^k$ input variables such that $\mathsf{KW}^+(f_{\mathsf{CanSearch}(\mathcal{C},g)})$ is equivalent to $\mathsf{CanSearch}(\mathcal{C}, g)$; to reduce clutter we will write $f_{\mathcal{C},g}$ instead of $f_{\mathsf{CanSearch}(\mathcal{C},g)}$. The function $f_{\mathcal{C},g}$ can be viewed as a monotone version of the CSP-SAT problem (as observed by [26, 35]), we refer to Appendix A for details.

3 Proof Outline

The rest of the paper is devoted to the proof of Theorem 1.1.

Theorem 1.1. *Let $\mathcal{C}$ be a constant-width unsatisfiable CNF on n variables, and let $\mathbf{F}$ be any field. For any good gadget g over $\mathbf{F}$ with rank$(g) = n^2$, the lifted function $f_{\mathcal{C},g}$ satisfies*

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = n^{\Theta(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))}.$$

Further, if $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \Theta(n)$, then for any good gadget g (of sufficiently large but constant rank),

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = 2^{\Theta(n)}.$$

In this section we give a technical overview of this result. Let us first state what it means for a gadget g to be good.

Definition 3.1. Let $\mathbf{F}$ be a field. A gadget $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ is *good* if $|\mathcal{X}| = O(\text{rank}(g))$ and for any matrices A, B over $\mathbf{F}$ of the same dimension

$$\text{rank}(\mathbf{1} \otimes A + g \otimes B) = \text{rank}(A) + \text{rank}(g)\text{rank}(B).$$

For the sake of concreteness, let $\mathbf{F}$ be a field, let $\mathcal{C}$ be an unsatisfiable k-CNF, and let $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ be a good gadget. Our goal is to relate Nullstellensatz refutations of $\mathcal{C}$ (under an appropriate encoding) to monotone span programs computing the function $f_{\mathcal{C},g}$.

The proof of Theorem 1.1 proceeds, broadly, in two steps: an upper bound and a lower bound. A hint that the upper bound holds can be seen by comparing Nullstellensatz refutations of $\mathcal{C}$ and algebraic tilings of $\mathsf{KW}^+(f_{\mathcal{C},g})$ side-by-side. To be even more concrete, let us suppose that $\text{char}(\mathbf{F}) \neq 2$ and we are using the encoding $\mathcal{E}^*(\mathcal{C})$ of the unsatisfiable CNF as a system of polynomials: then, a Nullstellensatz refutation of $\mathcal{E}^*(\mathcal{C})$ is given by a collection of polynomials $q_1, q_2, \ldots, q_m$ satisfying

$$\sum_{i=1}^{m} p_i q_i = 1, \tag{3}$$

where $p_i = \mathcal{E}^*(C_i)$ for each clause C_i and the *degree* of the refutation is $\max_i \deg(p_i q_i)$.

On the other hand, by the construction in the preliminaries, the Karchmer-Wigderson game $\mathsf{KW}^+(f_{\mathcal{C},g})$ is the same as the canonical search problem $\mathsf{CanSearch}(\mathcal{C}, g)$ from Definition 2.13. For the sake of simplicity, let us briefly suppose that the Karchmer-Wigderson game $\mathsf{KW}^+(f_{\mathcal{C},g})$ was instead equivalent to the

simpler search problem $\mathsf{Search}(\mathcal{C}, g)$, and therefore that the coordinate rectangles for the function $f_{\mathcal{C},g}$ were exactly the sets

$$X_\pi = \{(x, y) \in \mathcal{X}^n \times \mathcal{Y}^n \mid z = g^n(x, y) \text{ consistent with } \pi\}$$

for $\pi \in \mathsf{Cert}(\mathcal{C})$. In this simpler setting, an algebraic tiling of $\mathsf{KW}^+(f_{\mathcal{C},g})$ can be written as

$$\sum_{\pi \in \mathsf{Cert}(\mathcal{C})} A_\pi = \sum_{i=1}^{m} A_{\pi_i} = \mathbf{1}, \tag{4}$$

where A_π is embedded in X_π for each π, π_i is the unique certificate for the equation $p_i = 0$, and the size of the tiling is $\sum \text{rank}(A_{\pi_i})$.

To connect these two representations we use *pattern matrices*. Namely, if $p \in \mathbf{F}[z_1, z_2, \ldots, z_n]$ is a polynomial then we can compose p with the gadget g in the natural way to obtain a $\mathcal{X}^n \times \mathcal{Y}^n$ *pattern matrix* $p \circ g^n$ over $\mathbf{F}$ defined by

$$p \circ g^n(x, y) = p(g(x_1, y_1), g(x_2, y_2), \ldots, g(x_n, y_n)).$$

For our purposes, observe that for any clause C_i we have that $(p_i q_i) \circ g^n(x, y) \neq 0$ only if $z = g^n(x, y)$ is consistent with the certificate π_i of $p_i = 0$ — in other words, *the pattern matrix $(p_i q_i) \circ g^n$ is embedded in the coordinate rectangle X_{π_i} for each i.* Thus if $\sum_{i=1}^m p_i q_i = 1$ is a Nullstellensatz refutation of $\mathcal{E}^*(\mathcal{C})$ then $\sum_{i=1}^m p_i q_i \circ g^n = \mathbf{1}$ is an algebraic tiling of $\mathsf{KW}^+(f_{\mathcal{C},g})$!

A problem with the above argument is that it is not obvious if the size of the algebraic tiling $\sum_{i=1}^m \text{rank}(p_i q_i \circ g^n)$ is related to the degree of underlying Nullstellensatz refutation $\max_{i \in [m]} \deg(p_i q_i)$. In order to guarantee this, we need to choose the gadget g so that for every polynomial p, the degree of p is directly related to the rank of the pattern matrix $p \circ g^n$. In the case that the polynomial p is real-valued, such a gadget was (implicitly) shown to exist by Sherstov [43].

Theorem 3.2 (Corollary of Theorem 4.3 in [43])**.** *Let p be a real multilinear polynomial over n variables $z_1, z_2, \ldots, z_n$. For each $\lambda \in \mathbf{Z}_+$ there is a gadget g_λ such that*

$$\textit{rank}_{\mathbf{R}}(p \circ g_\lambda^n) = \sum_{S : \hat{p}(S) \neq 0} \lambda^{|S|}.$$

Using the gadget g_λ from the previous theorem would immediately yield the upper bound of Theorem 1.1 when $\mathbf{F} = \mathbf{R}$ by following the proof sketch above. However, there is a problem in trying to generalize the proof of Theorem 3.2 to arbitrary fields: in [43] the singular values of $p \circ g_\lambda^n$ are exactly computed for every real polynomial p, and the singular value decomposition becomes quite useful for other results in that work. Since singular values are not well-defined over finite fields this proof cannot be directly generalized.

One of the main contributions of the present paper, which we hope will have other applications, is the following strengthening of Theorem 3.2 to all fields.

Theorem 1.2. *Let $p \in \mathbf{F}[z_1, z_2, \ldots, z_n]$ be a multilinear polynomial and let $\mathbf{F}$ be a field. For any good gadget $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ we have*

$$\textit{rank}_{\mathbf{F}}(p \circ g^n) = \sum_{S : \hat{p}(S) \neq 0} \textit{rank}_{\mathbf{F}}(g)^{|S|}$$

where $\hat{p}(S)$ denotes the coefficient of the monomial $\prod_{i \in S} z_i$ in p.

We note that from this theorem we can recover Theorem 3.2 directly: Sherstov's gadget g_λ from the statement of Theorem 3.2 is good for all fields with $\text{char}(\mathbf{F}) \neq 2$ and also satisfies $\text{rank}_{\mathbf{R}}(g_\lambda) = \lambda$. Using Theorem 1.2 instead of Theorem 3.2, along with the proof sketch above (suitably modified using the search problem $\mathsf{CanSearch}(\mathcal{C}, g)$ instead of $\mathsf{Search}(\mathcal{C}, g)$), yields the upper bound in Theorem 1.1:

Theorem 3.3. *Let n, k be positive integers and let $\mathcal{C}$ be an unsatisfiable k-CNF with n variables and m clauses. Let $\mathbf{F}$ be a field and let g be a good gadget with rank$(g) = O(\mathsf{poly}(n))$ over $\mathbf{F}$. Then*

$$\chi_{\mathbf{F}}(f_{\mathcal{C},g}) \leq mn^{O(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))}.$$

Furthermore, suppose $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) \geq \varepsilon n$ for some constant ε independent of n. Then for any good gadget g over $\mathbf{F}$ with rank$(g) > 2^{1/\varepsilon}$,

$$\chi_{\mathbf{F}}(f_{\mathcal{C},g}) \leq m2^{O(n)}.$$

Next let us discuss the lower bound in Theorem 1.1. In principle, a direct proof would proceed by taking an algebraic tiling $A_1, A_2, \ldots, A_N$ of $\mathsf{KW}^+(f_{\mathcal{C},g})$ of size $\chi_{\mathbf{F}}(f_{\mathcal{C},g})$ and then extracting a Nullstellensatz refutation of the underlying system of polynomial equations in $\mathcal{C}$ with degree roughly $O(\log \chi_{\mathbf{F}}(f_{\mathcal{C},g})/\log n)$; indeed, this is the approach that the other lifting theorems in the literature tend to follow [16, 25, 33, 39]. In particular, approaching the problem this way seems to require extracting a polynomial q_i from each matrix A_i in the tiling such that $\deg(q_i) \approx \log \mathrm{rank}(A_i)/\log n$. Since the tiling we begin with is chosen arbitrarily it does not have the structure of a pattern matrix, and so it is not clear how to go about extracting such a polynomial.

We deviate from this approach, and instead prove a different lifting theorem from the *algebraic gap complexity* $\mathsf{gap}_{\mathbf{F}}(\mathcal{C})$ to *Razborov's rank measure* $\mu_{\mathbf{F}}(f_{\mathcal{C},g})$ — this allows us to exploit the structure of pattern matrices and Theorem 1.2.

Theorem 3.4. *Let n, k be positive integers and let $\mathcal{C}$ be an unsatisfiable k-CNF over n variables. Let $\mathbf{F}$ be a field and let g be a good gadget with rank$(g) = n^2$ over $\mathbf{F}$. Then*

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) \geq \Omega(n^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}).$$

Furthermore, suppose $\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) \geq \varepsilon n$ for some $\varepsilon > 0$. Then for any good gadget g over $\mathbf{F}$ with rank$(g) > 2^{1/\varepsilon}$,

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) \geq 2^{\Omega(n)}.$$

A special case of this theorem was proven in [42] when the field $\mathbf{F}$ is the real numbers: indeed, the proof of the special case crucially relied on the "real rank-lifting" Theorem 3.2. We obtain Theorem 3.4 by following the proof from [42] while replacing each application of Theorem 3.2 with the more general Theorem 1.2. However, we note that this lower bound does *not* immediately imply the lower bound in Theorem 1.1 since it is in terms of the algebraic gap complexity and not Nullstellensatz degree. To obtain Theorem 1.1 we show that the algebraic gap complexity and the Nullstellensatz degree are the same for unsatisfiable CNFs.

Theorem 3.5. *For any unsatisfiable CNF $\mathcal{C}$ and any field $\mathbf{F}$ we have $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \mathsf{gap}_{\mathbf{F}}(\mathcal{C})$.*

We have abused notation in the statement of Theorem 3.5, as $\mathsf{gap}_{\mathbf{F}}$ is defined for polynomial systems of equations and not CNFs. Theorem 3.5 turns out be quite sensitive to the encoding of unsatisfiable CNFs $\mathcal{C}$ as polynomial systems of equations — in particular, we can only prove $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C}))$ when $\mathrm{char}(\mathcal{C}) \neq 2$ and $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \mathsf{gap}_{\mathbf{F}}(\mathcal{E}(\mathcal{C}))$ when $\mathrm{char}(\mathcal{C}) = 2$. As a result, we define

$$\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) = \begin{cases} \mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C})) & \text{if } \mathrm{char}(\mathbf{F}) \neq 2 \\ \mathsf{gap}_{\mathbf{F}}(\mathcal{E}(\mathcal{C})) & \text{if } \mathrm{char}(\mathbf{F}) = 2. \end{cases}$$

We conjecture that $\mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C})) \neq \mathsf{gap}_{\mathbf{F}}(\mathcal{E}(C))$ in general for fields of characteristic other than 2, and note that this is somewhat remarkable since Nullstellensatz degree is easily seen to be independent of the encoding.

The rest of the paper is organized as follows. In Section 4 we prove our main degree-to-rank connection, Theorem 1.2, and construct two families of good gadgets g. Using this construction, we prove Theorem 3.4 in Section 4.2. In Section 5 we prove Theorem 3.5, showing algebraic gaps and Nullstellensatz are dual. Finally in Section 6 we prove Theorem 3.3 and then Theorem 1.1 follows as an easy corollary. The main applications are proved in the Appendix.

4 Rank Lifting Over All Fields

In this section we prove Theorem 1.2, which is our general degree-to-rank lifting theorem, and then Theorem 3.4, which lifts algebraic gap complexity $\mathsf{gap}_{\mathbf{F}}$ to the rank measure $\mu_{\mathbf{F}}$.

4.1 Proof of Theorem 1.2 and Gadget Construction

As discussed above, Theorem 1.2 is a generalization of a result of Sherstov, which holds for real polynomials and only for a particular choice of gadget g. We generalize the "real rank-lifting theorem" to work over all fields, and also give a general sufficient property on gadgets g for which such a degree-to-rank lift is possible. In fact, this general property is satisfied by Sherstov's gadget for all fields of characteristic other than 2, which we prove after proving Theorem 1.2.

The proof of Theorem 1.2 is elementary, using induction and basic algebraic properties of the Kronecker product. In contrast, the special case of Theorem 1.2 for real polynomials and a special gadget g [43, Theorem 4.3] follows from an explicit calculation of the singular values of $p \circ g^n$.

Theorem 1.2. *Let $p \in \mathbf{F}[z_1, z_2, \ldots, z_n]$ be a multilinear polynomial and let $\mathbf{F}$ be a field. For any good gadget $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ we have*

$$\mathit{rank}_{\mathbf{F}}(p \circ g^n) = \sum_{S : \hat{p}(S) \neq 0} \mathit{rank}_{\mathbf{F}}(g)^{|S|}$$

where $\hat{p}(S)$ denotes the coefficient of the monomial $\prod_{i \in S} z_i$ in p.

Proof. Suppose that A is an $m \times n$ matrix and B is a $p \times q$ matrix over $\mathbf{F}$. Recall from the preliminaries that if we think of A and B as mappings $A : [m] \times [n] \to \mathbf{F}$, $B : [p] \times [q] \to \mathbf{F}$, then the Kronecker product has a natural interpretation as the mapping $A \otimes B : ([m] \times [p]) \times ([n] \times [q]) \to \mathbf{F}$ defined by

$$(A \otimes B)((i,k),(j,\ell)) = A(i,j)B(k,\ell). \tag{5}$$

From this fact we have the following claim.

Claim. Let $S \subseteq [n]$, and let $z_S = \prod_{i \in S} z_i$ denote a monomial over $\mathbf{F}[z_1, z_2, \ldots, z_n]$. Then

$$z_S \circ g^n = \bigotimes_{i=1}^{n} M_S(i)$$

where $M_S(i) = g$ if $i \in S$ and $M_S(i) = \mathbf{1}$ otherwise.

Proof of Claim. For notational simplicity suppose that $S = \{1, 2, \ldots, t\}$ for some $t \leq n$, and a symmetric

calculation applies for general S. Then

$$\begin{aligned}
z_S \circ g^n &= [z_S(g(x_1,y_1), g(x_2,y_2), \cdots, g(x_n,y_n))]_{(x,y)\in\mathcal{X}^n\times\mathcal{Y}^n} \\
&= \left[\prod_{i\in S} g(x_i,y_i)\right]_{(x,y)\in\mathcal{X}^n\times\mathcal{Y}^n} \\
&= [g(x_1,y_1)g(x_2,y_2)\cdots g(x_t,y_t)\mathbf{1}(x_{t+1},y_{t+1})\cdots\mathbf{1}(x_n,y_n)]_{(x,y)\in\mathcal{X}^n\times\mathcal{Y}^n} \\
&= \underbrace{g\otimes g\otimes\cdots\otimes g}_{t \text{ times}} \otimes \underbrace{\mathbf{1}\otimes\cdots\otimes\mathbf{1}}_{n-t \text{ times}} \\
&= \bigotimes_{i=1}^{n} M_S(i),
\end{aligned}$$

where we have used Equation 5. □

We need the following useful properties of the Kronecker product.

1. For any matrices A and B taking values over $\mathbf{F}$,

$$\text{rank}_{\mathbf{F}}(A\otimes B) = \text{rank}_{\mathbf{F}}(A)\text{rank}_{\mathbf{F}}(B).$$

2. The Kronecker product is *bilinear*: if A, B, C, D are matrices then

$$(A+B)\otimes C = A\otimes C + B\otimes C$$

and

$$A\otimes(C+D) = A\otimes C + A\otimes D$$

whenever the sums are well-defined.

Using these properties and the claim, we prove the lemma by induction on n. Recall from Definition 3.1 that a gadget $g : \mathcal{X}\times\mathcal{Y}\to\mathbf{F}$ is good if for any matrices A, B over $\mathbf{F}$ of the same dimension

$$\text{rank}(\mathbf{1}\otimes A + g\otimes B) = \text{rank}(A) + \text{rank}(g)\text{rank}(B).$$

When $n = 0$ the polynomial p is just a constant in $\mathbf{F}$, and the matrix $p\circ g^0$ is the 1×1 matrix $[\hat{p}(\emptyset)]$. In this case the conclusion of the theorem is trivially satisfied — if $\hat{p}(\emptyset) = 0$ then $\text{rank}(p\circ g^0) = 0$ and if $\hat{p}(\emptyset)\neq 0$ then $\text{rank}(p\circ g^0) = 1 = \text{rank}(g)^0$.

Now, suppose that $n \geq 0$, and write $p = q + z_1 r$, where $q, r \in \mathbf{F}[z_2, z_3, \ldots, z_n]$. By the claim and the bilinearity of the Kronecker product, we can write

$$\begin{aligned}
p\circ g^n &= \sum_{S:\hat{p}(S)\neq 0} \hat{p}(S)\bigotimes_{i=1}^{n} M_S(i) \\
&= \sum_{\substack{S:\hat{p}(S)\neq 0 \\ 1\notin S}} \hat{p}(S)\cdot\mathbf{1}\otimes\bigotimes_{i=2}^{n} M_S(i) + \sum_{\substack{S:\hat{p}(S)\neq 0 \\ 1\in S}} \hat{p}(S)\cdot g\otimes\bigotimes_{i=2}^{n} M_S(i) \\
&= \mathbf{1}\otimes\left(\sum_{\substack{S:\hat{p}(S)\neq 0 \\ 1\notin S}} \hat{p}(S)\bigotimes_{i=2}^{n} M_S(i)\right) + g\otimes\left(\sum_{\substack{S:\hat{p}(S)\neq 0 \\ 1\in S}} \hat{p}(S)\bigotimes_{i=2}^{n} M_S(i)\right) \\
&= \mathbf{1}\otimes(q\circ g^{n-1}) + g\otimes(r\circ g^{n-1}).
\end{aligned}$$

Therefore, applying the inductive assumption, we have

$$\begin{aligned}\text{rank}_{\mathbf{F}}(p \circ g^n) &= \text{rank}_{\mathbf{F}}(\mathbf{1} \otimes (q \circ g^{n-1}) + g \otimes (r \circ g^{n-1})) \\ &= \text{rank}_{\mathbf{F}}(q \circ g^{n-1}) + \text{rank}_{\mathbf{F}}(g)\text{rank}_{\mathbf{F}}(r \circ g^{n-1}) \\ &= \sum_{T:\hat{q}(T)\neq 0} \text{rank}_{\mathbf{F}}(g)^{|T|} + \sum_{T:\hat{r}(T)\neq 0} \text{rank}_{\mathbf{F}}(g)^{|T|+1} \\ &= \sum_{\substack{S:\hat{p}(S)\neq 0 \\ 1\notin S}} \text{rank}_{\mathbf{F}}(g)^{|S|} + \sum_{\substack{S:\hat{p}(S)\neq 0 \\ 1\in S}} \text{rank}_{\mathbf{F}}(g)^{|S|} = \sum_{S:\hat{p}(S)\neq 0} \text{rank}_{\mathbf{F}}(g)^{|S|},\end{aligned}$$

where we note that in the above sums $T \subseteq \{2, 3, \ldots, n\}$ and $S \subseteq [n]$. □

In the rest of the subsection, we describe two families of good gadgets: one that will be used for fields of characteristic other than 2 and the other that will be used for characteristic 2. First, for characteristic other than 2, we use the following gadget (originally introduced by Sherstov [43], and also used by Göös and Pitassi [26] and [36, 42]).

Definition 4.1. For any positive integer λ, define the gadget $g_\lambda : ([\lambda] \times \{-1, 1\}) \times \{-1, 1\}^\lambda \to \{-1, 1\}$ by

$$g_\lambda((x, b), y) = by_x.$$

Lemma 4.2. *Let λ be a positive integer and let $\mathbf{F}$ be a field with characteristic other than 2. Then g_λ is good and satisfies rank$(g_\lambda) = \lambda$.*

Proof. To see that $\text{rank}_{\mathbf{F}}(g_\lambda) = \lambda$ it will be helpful to describe the structure of the matrix g_λ. Let $x \in [\lambda]$ and $b \in \{\pm 1\}$. By definition of g_λ it should be clear that g_λ is equivalent (up to a permutation of rows) to the matrix

$$\begin{pmatrix} F_\lambda \\ -F_\lambda \end{pmatrix}$$

where $F_\lambda : [\lambda] \times \{\pm 1\}^\lambda$ is defined by $F_\lambda(x, y) = y_x$. Note that F_λ is full rank (that is, it has rank λ), implying that $\text{rank}_{\mathbf{F}}(g_\lambda) = \text{rank}_{\mathbf{F}}(F_\lambda) = \lambda$.

By the definition of the Kronecker product

$$\mathbf{1}_{2\lambda,2^\lambda} \otimes A + g \otimes B = \mathbf{1}_{2\lambda,2^\lambda} \otimes A + \begin{pmatrix} F_\lambda \otimes B \\ -F_\lambda \otimes B \end{pmatrix} = \begin{pmatrix} 1_{\lambda,2^\lambda} \otimes A + F_\lambda \otimes B \\ 1_{\lambda,2^\lambda} \otimes A - F_\lambda \otimes B \end{pmatrix}.$$

By adding the top half of the resulting matrix to the bottom half, we obtain

$$\begin{pmatrix} 1_{\lambda,2^\lambda} \otimes A + F_\lambda \otimes B \\ 2 \cdot 1_{\lambda,2^\lambda} \otimes A \end{pmatrix},$$

from which we get the matrix

$$\begin{pmatrix} F_\lambda \otimes B \\ 1_{\lambda,2^\lambda} \otimes A \end{pmatrix}$$

after dividing the bottom half by 2 and then subtracting it from the top half. Since the Kronecker product is multiplicative with respect to rank we have that $\text{rank}(F_\lambda \otimes B) = \lambda \cdot \text{rank}(B)$, and thus there exists a sequence of row operations that can be applied to the matrix $F_\lambda \otimes B$ to obtain the matrix $(I_{\lambda \cdot \text{rank}(B)}\ \mathbf{0})$ where $\mathbf{0}$ is a block matrix of 0s. Applying these row operations to the top half of the previous matrix we obtain

$$\begin{pmatrix} I_{\lambda \cdot \text{rank}(B)} & \mathbf{0} \\ 1_{\lambda,2^{\lambda-1}} \otimes A & 1_{\lambda,2^{\lambda-1}} \otimes A \end{pmatrix}.$$

The rank of this matrix is clearly $\text{rank}_{\mathbf{F}}(A) + \lambda \cdot \text{rank}_{\mathbf{F}}(B) = \text{rank}_{\mathbf{F}}(A) + \text{rank}_{\mathbf{F}}(g_\lambda)\text{rank}_{\mathbf{F}}(B)$. □

We note that one can remove some unnecessary rows and columns from the gadget g_λ to obtain a smaller gadget that is also good with the same rank. Next, we introduce a new good gadget that will be used for characteristic 2.

Definition 4.3. For any positive integer λ define $h_\lambda : [\lambda+1] \times [\lambda+1] \to \{0,1\}$ by

$$h_\lambda(x,y) = \begin{cases} 1 & \text{if } x = y = i \text{ for some } i \in [\lambda] \\ 0 & \text{otherwise.} \end{cases}$$

That is, h_λ is the $(\lambda+1) \times (\lambda+1)$ identity matrix with one of the 1s deleted.

Lemma 4.4. *Let λ be a positive integer and let $\mathbf{F}$ be any field. Then h_λ is good and satisfies rank$(h_\lambda) = \lambda$.*

Proof. Clearly rank$(h_\lambda) = \lambda$, so we focus on the linearity property of rank. By definition of $\mathbf{1}$, h_λ, and the Kronecker product, we have

$$\mathbf{1} \otimes A = \begin{pmatrix} A & A & \cdots & A \\ A & A & \cdots & A \\ & & \vdots & \\ A & A & \cdots & A \end{pmatrix} \quad h_\lambda \otimes B = \begin{pmatrix} B & 0 & \cdots & 0 & 0 \\ 0 & B & \cdots & 0 & 0 \\ & & \vdots & & \\ 0 & 0 & \cdots & B & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix}.$$

Adding them yields the matrix

$$\begin{pmatrix} A+B & A & \cdots & A & A \\ A & A+B & \cdots & A & A \\ & & \vdots & & \\ A & A & \cdots & A+B & A \\ A & A & \cdots & A & A \end{pmatrix},$$

which is easily verified to be row- and column-equivalent to

$$\begin{pmatrix} B & 0 & \cdots & 0 & 0 \\ 0 & B & \cdots & 0 & 0 \\ & & \vdots & & \\ 0 & 0 & \cdots & B & 0 \\ 0 & 0 & \cdots & 0 & A \end{pmatrix},$$

by subtracting the final column from each other column, and then subtracting the last row from all other rows. The rank property follows since there are λ copies of B on the diagonal. □

4.2 Lifting Algebraic Gaps to the Rank Measure

Next, using Theorem 1.2 we show how to lift algebraic gap complexity to the rank measure over all fields. In the statement of Theorem 3.4 below we abuse notation and write $\mathsf{gap}_{\mathbf{F}}(\mathcal{C})$ for an unsatisfiable CNF $\mathcal{C}$ to mean $\mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C}))$ when $\mathrm{char}(\mathbf{F}) \neq 2$ and $\mathsf{gap}_{\mathbf{F}}(\mathcal{E}(\mathcal{C}))$ when $\mathrm{char}(\mathbf{F}) = 2$. The proof of the theorem is actually insensitive to the details of the encoding $\mathcal{C}$ as a system of polynomial equations: our choice of encodings is made due to Theorem 3.5.

Theorem 3.4. *Let n, k be positive integers and let $\mathcal{C}$ be an unsatisfiable k-CNF over n variables. Let $\mathbf{F}$ be a field and let g be a good gadget with rank$(g) = n^2$ over $\mathbf{F}$. Then*

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) \geq \Omega(n^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}).$$

Furthermore, suppose $\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) \geq \varepsilon n$ for some $\varepsilon > 0$. Then for any good gadget g over $\mathbf{F}$ with rank$(g) > 2^{1/\varepsilon}$,

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) \geq 2^{\Omega(n)}.$$

Proof. Let g be a good gadget over $\mathbf{F}$ (for concreteness, one can think of g as being g_λ if $\mathrm{char}(\mathbf{F}) \neq 2$ and g as h_λ if $\mathrm{char}(\mathbf{F}) = 2$). Let $p \in \mathbf{F}[z_1, z_2, \ldots, z_n]$ be the polynomial witnessing the algebraic gap complexity $\mathsf{gap}_{\mathbf{F}}(\mathcal{C})$, and let $A = p \circ g^n$ be the $\mathcal{X}^n \times \mathcal{Y}^n$ matrix generated by composing p and g. First suppose that $\mathrm{rank}(g) = n^2$, and we prove

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}, A) = \frac{\mathrm{rank}_{\mathbf{F}}(A)}{\max\limits_{X_{\pi,\alpha}} \mathrm{rank}_{\mathbf{F}}(A\restriction X_{\pi,\alpha})} \geq \Omega(n^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}).$$

The numerator is easy to bound. Since g is good, by Theorem 1.2 we have

$$\mathrm{rank}_{\mathbf{F}}(A) = \sum_{S:\hat{p}(S)\neq 0} \mathrm{rank}(g)^{|S|} \geq \mathrm{rank}(g)^n$$

since $\deg p = n$ by the definition of algebraic gaps. For the denominator, let $X_{\pi,\alpha}$ be an arbitrary rectangle from $\mathsf{CanSearch}(\mathcal{C}, g)$. We show that

$$\mathrm{rank}_{\mathbf{F}}(A\restriction X_{\pi,\alpha}) = \sum_{S:\widehat{p\restriction\pi}(S)\neq 0} \mathrm{rank}(g)^{|S|}. \tag{6}$$

To see Equation 6, we claim that the matrix $A\restriction X_{\pi,\alpha}$ is column-equivalent to the block matrix

$$[(p\restriction\pi) \circ g^{[n]\setminus\mathsf{vars}(\pi)}), (p\restriction\pi) \circ g^{[n]\setminus\mathsf{vars}(\pi)}), \ldots, (p\restriction\pi) \circ g^{[n]\setminus\mathsf{vars}(\pi)})]$$

for some number of copies of the matrix $(p\restriction\pi) \circ g^{[n]\setminus\mathsf{vars}(\pi)}$. Equation 6 immediately follows as the claim implies that

$$\mathrm{rank}_{\mathbf{F}}(A\restriction X_{\pi,\alpha}) = \mathrm{rank}_{\mathbf{F}}((p\restriction\pi) \circ g^{[n]\setminus\mathsf{vars}(\pi)}) = \sum_{S:\widehat{p\restriction\pi}(S)\neq 0} \mathrm{rank}(g)^{|S|}$$

by Theorem 1.2. So let us prove the claim.

By the definition of $\mathsf{CanSearch}(\mathcal{C}, g)$, for all $(x, y) \in X_{\pi,\alpha}$ we have that $g^{\mathsf{vars}(\pi)}(x_{\mathsf{vars}(\pi)}, y_{\mathsf{vars}(\pi)}) = \pi$ and $x_{\mathsf{vars}(\pi)} = \alpha$. Let us first fix any assignment β to $y_{\mathsf{vars}(\pi)}$ so that $g^{\mathsf{vars}(\pi)}(\alpha, \beta) = \pi$. Then for all $(x, y) \in X_{\pi,\alpha}$ such that $y_{\mathsf{vars}(\pi)} = \beta$ we have

$$\begin{aligned} g^n(x, y) &= g^{\mathsf{vars}(\pi)}(\alpha, \beta) g^{[n]\setminus\mathsf{vars}(\pi)}(x_{[n]\setminus\mathsf{vars}(\pi)}, y_{[n]\setminus\mathsf{vars}(\pi)}) \\ &= \pi g^{[n]\setminus\mathsf{vars}(\pi)}(x_{[n]\setminus\mathsf{vars}(\pi)}, y_{[n]\setminus\mathsf{vars}(\pi)}), \end{aligned}$$

thus ranging $x_{[n]\setminus\mathsf{vars}(\pi)}, y_{[n]\setminus\mathsf{vars}(\pi)}$ over all values yields the matrix $(p\restriction\pi) \circ g^{[n]\setminus\mathsf{vars}(\pi)}$. Ranging $y_{\mathsf{vars}(\pi)}$ over all β such that $g^{\mathsf{vars}(\pi)}(\alpha, \beta) = \pi$ yields the claim and Equation 6.

By Equation 6, we have

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}, A) = \frac{\sum\limits_{S:\hat{p}(S)\neq 0} \mathrm{rank}(g)^{|S|}}{\max\limits_{\pi\in\mathsf{Cert}(\mathcal{C})} \sum\limits_{S:\widehat{p\restriction\pi}(S)\neq 0} \mathrm{rank}(g)^{|S|}} \geq \frac{\mathrm{rank}(g)^n}{\max\limits_{\pi\in\mathsf{Cert}(\mathcal{C})} \sum\limits_{S:\widehat{p\restriction\pi}(S)\neq 0} \mathrm{rank}(g)^{|S|}}$$

where the inequality follows since $\deg p = n$. Since p witnesses the algebraic gap of $\mathcal{C}$, we have that $\deg p\restriction_\pi \leq n - \mathsf{gap}_{\mathbf{F}}(\mathcal{C})$ for all $\pi \in \mathsf{Cert}(\mathcal{C})$. We may clearly assume that $\hat{p}(S) = 0$ when $|S| < n - \mathsf{gap}_{\mathbf{F}}(\mathcal{C})$, and so let us first set $\mathrm{rank}(g) = n^2$. Then for any $\pi \in \mathsf{Cert}(\mathcal{C})$

$$\begin{aligned}
\sum_{S:\widehat{p\restriction_\pi}(S)\neq 0} \mathrm{rank}(g)^{|S|} &\leq \sum_{i=0}^{k} \binom{n}{\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) - i} n^{2(n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})-i)} \\
&\leq \sum_{i=0}^{k} \left(\frac{en}{\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) - i}\right)^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})-i} n^{2(n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})-i)} \\
&\leq \sum_{i=0}^{k} \left(\frac{e}{\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) - i}\right)^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})-i} n^{2n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})-i} \\
&\leq n^{2n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})} \sum_{i=0}^{k} \left(\frac{e}{\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) - i}\right)^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})-i} \leq 6n^{2n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}
\end{aligned}$$

since $e + (e/2)^2 + (e/3)^3 + \cdots \leq 6$. Putting it all together, we get

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}, A) \geq \frac{n^{2n}}{6n^{2n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}} = cn^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}$$

where $c = 1/6$, proving the first part of the theorem.

Next we prove the second part of the theorem. Assume $\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) \geq \varepsilon n$ and $\mathrm{rank}(g) > 2^{1/\varepsilon}$. By Equation 6,

$$\sum_{S:\widehat{p\restriction_\pi}(S)\neq 0} \mathrm{rank}(g)^{|S|} \leq \sum_{i=0}^{k} \binom{n}{\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) - i} \cdot \mathrm{rank}(g)^{n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})-i} \leq 2^n \cdot \mathrm{rank}(g)^{n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}.$$

Then

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}, A) \geq \frac{\mathrm{rank}(g)^n}{2^n \cdot \mathrm{rank}(g)^{n-\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}} = \frac{\mathrm{rank}(g)^{\mathsf{gap}_{\mathbf{F}}(\mathcal{C})}}{2^n} \geq 2^{(\varepsilon \log \mathrm{rank}(g) - 1)n} = 2^{\Omega(n)}. \qquad \square$$

5 Algebraic Gaps and Nullstellensatz

In this section we prove Theorem 3.5, which we restate here for convenience.

Theorem 3.5. *For any unsatisfiable CNF $\mathcal{C}$ and any field $\mathbf{F}$ we have $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \mathsf{gap}_{\mathbf{F}}(\mathcal{C})$.*

The proof uses the dual characterization of Nullstellensatz degree by *d-designs* [12, 14]. Let $\mathbf{F}$ be a field of characteristic other than 2, and let $\mathcal{P}$ be an unsatisfiable system of multilinear polynomial equations over $\mathbf{F}[z_1, z_2, \ldots, z_n]$. A d-design for $\mathcal{P}$ is a linear functional D on the space of polynomials satisfying the following axioms:

1. $D(1) = 1$.
2. For all $P \in \mathcal{F}$ and all polynomials Q such that $\deg(PQ) \leq d$, we have $D(PQ) = 0$.
3. $D(z^2 P) = D(zP)$ for all variables z and all polynomials P of degree less than $d - 1$.

It is known (see, for example, [12]) that the system $\mathcal{P}$ does not have a Nullstellensatz refutation of degree d if and only if it has a d-design, and thus every system of polynomial equations $\mathcal{F}$ has a $(\mathsf{NS}(\mathcal{F})-1)$-design.

We prove Theorem 3.5 in two steps: one for characteristic 2 and the other for characteristic different than 2 (although, the proofs are essentially the same). Before we begin, we will need the following easy lemma regarding the *dual* of a CNF. Let $\mathcal{C}$ be an unsatisfiable CNF. For any clause $C \in \mathcal{C}$ let $C^\dagger$ denote the clause obtained by negating every literal in C (so, z is replaced with $\neg z$ and $\neg z$ is replaced with z). Let $\mathcal{C}^\dagger$ be the CNF obtained from $\mathcal{C}$ by replacing each clause in $\mathcal{C}$ with its dual, and note that $\mathcal{C}^\dagger$ is unsatisfiable if and only if $\mathcal{C}$ is unsatisfiable.

Lemma 5.1. *For any field* $\mathbf{F}$, $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \mathsf{NS}_{\mathbf{F}}(\mathcal{C}^\dagger)$.

Proof. Let $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$ be an unsatisfiable CNF over variables $z_1, z_2, \ldots, z_n$. We prove $\mathsf{NS}_{\mathbf{F}}(\mathcal{E}(\mathcal{C})) = \mathsf{NS}_{\mathbf{F}}(\mathcal{E}(\mathcal{C}^\dagger))$, and note that $\mathsf{NS}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C})) = \mathsf{NS}_{\mathbf{F}}(\mathcal{E}(\mathcal{C}))$ over every field. It will be convenient to consider the following alternative encoding of CNFs $\mathcal{C}$ as a system of polynomial equations. For each variable z_i introduce two variables, denoted z_i and $\overline{z_i}$, along with the axioms

$$\forall i : z_i(1 - z_i) = 0, \quad z_i + \overline{z_i} = 1$$

which enforce that $z_i = 1 - \overline{z_i}$ and $z_i, \overline{z_i} \in \{0, 1\}$ (this encoding is typically used in the "polynomial calculus with resolution", or PCR, proof system). Then encode each clause C_i as

$$\mathcal{E}^{\square}(C_i) = \prod_{j \in C_i^+} z_j \prod_{j \in C_i^-} \overline{z_i},$$

which yields an encoding of $\mathcal{C}$ in $\mathbf{F}[z_1, \overline{z_1}, z_2, \overline{z_2}, \ldots, z_n, \overline{z_n}]$. We show that $\mathsf{NS}_{\mathbf{F}}(\mathcal{E}(\mathcal{C})) = \mathsf{NS}_{\mathbf{F}}(\mathcal{E}^{\square}(\mathcal{C}))$ and then that $\mathsf{NS}_{\mathbf{F}}(\mathcal{E}^{\square}(\mathcal{C})) = \mathsf{NS}_{\mathbf{F}}(\mathcal{E}^{\square}(\mathcal{C}^\dagger))$.

First observe that $\mathsf{NS}_{\mathbf{F}}(\mathcal{E}(\mathcal{C})) \leq \mathsf{NS}_{\mathbf{F}}(\mathcal{E}^{\square}(\mathcal{C}))$ is easy: in the refutation of $\mathcal{E}^{\square}(\mathcal{C})$ replace every literal z_i with z_i and every literal $\overline{z_i}$ with $1 - z_i$. So, we focus on proving $\mathsf{NS}_{\mathbf{F}}(\mathcal{E}^{\square}(\mathcal{C})) \leq \mathsf{NS}_{\mathbf{F}}(\mathcal{E}(\mathcal{C}))$.

Suppose we have a Nullstellensatz refutation of $\mathcal{E}(\mathcal{C})$, and we construct a Nullstellensatz refutation of $\mathcal{E}^{\square}(\mathcal{C})$ of the same degree. For this, it suffices to show that there is a low degree proof of $\mathcal{E}(C)$ from $\mathcal{E}^{\square}(\mathcal{C})$ for each clause $C \in \mathcal{C}$. Write $\mathcal{E}^{\square}(C)$ as $\prod_{i \in C^+} z_i \prod_{i \in C^-} \overline{z_i}$, and we use the axioms $\overline{z_j} + z_j - 1 = 0$ for each $j \in C^+$ to derive $\mathcal{E}(C)$. To do this, multiply the axiom by $-\prod_{i \in C^-} \overline{z_i}$, yielding

$$-\prod_{i \in C^-} \overline{z_i}(\overline{z_j} + z_j - 1) = (1 - \overline{z_j}) \prod_{i \in C^-} \overline{z_i} - z_j \prod_{i \in C^-} \overline{z_i}.$$

Doing this for each $i \in C^+$ and factoring yields

$$\prod_{j \in C^+} (1 - \overline{z_j}) \prod_{j \in C^-} \overline{z_i} - \prod_{j \in C^+} z_j \prod_{j \in C^-} \overline{z_i}$$

which yields $\mathcal{E}(C)$ (over $\overline{z_i}$ variables) after adding $\mathcal{E}^{\square}(C)$. Performing this multiplication for each $C \in \mathcal{C}$ yields $\mathcal{E}(\mathcal{C})$, and it is easy to see that the degree is less than the degree of $\mathcal{E}(\mathcal{C})$.

Now let us prove $\mathcal{E}^{\square}(\mathcal{C}) = \mathcal{E}^{\square}(\mathcal{C}^\dagger)$. Observe that if $\sum_{C \in \mathcal{C}} \mathcal{E}^{\square}(C) q_i = 1$ is a Nullstellensatz refutation of $\mathcal{E}^{\square}(\mathcal{C})$ then $\sum_{C \in \mathcal{C}} \mathcal{E}^{\square}(C^\dagger) q_i^\dagger = 1$ is a Nullstellensatz refutation of $\mathcal{C}^\dagger$, where $q_i^\dagger$ is the polynomial obtained from q_i by exchanging the variables z_i and $\overline{z_i}$ for each $i \in [n]$ and $b \in \{0, 1\}$. This shows that $\mathsf{NS}_{\mathbf{F}}(\mathcal{E}^{\square}(\mathcal{C})) = \mathsf{NS}_{\mathbf{F}}(\mathcal{E}^{\square}(\mathcal{C}^\dagger))$, and thus $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \mathsf{NS}_{\mathbf{F}}(\mathcal{C}^\dagger)$. □

Using this proposition we are ready to prove Theorem 3.5. First we prove the theorem for fields of characteristic other than 2.

Lemma 5.2. *Let* $\mathbf{F}$ *be a field of characteristic other than 2 and let* $\mathcal{C}$ *be an unsatisfiable* k*-CNF. Then* $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C}))$.

Proof. Proposition 2.10 shows that $\mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C})) \leq \mathsf{NS}_{\mathbf{F}}(\mathcal{C})$, so, we show that $\mathsf{gap}_{\mathbf{F}}(\mathcal{E}^*(\mathcal{C})) \geq \mathsf{NS}_{\mathbf{F}}(\mathcal{C})$ (note that the Nullstellensatz degree of $\mathcal{C}$ is independent of the encoding). We show that if $\mathcal{E}^*(\mathcal{C}^\dagger)$ has a d-design then $\mathcal{E}^*(\mathcal{C})$ has algebraic gap complexity at least $d+1$. By Lemma 5.1, $\mathcal{E}^*(\mathcal{C}^\dagger)$ has an $(\mathsf{NS}_{\mathbf{F}}(\mathcal{C})-1)$-design, and so this completes the proof of the lemma.

So, let D be a d-design for $\mathcal{E}^*(\mathcal{C}^\dagger)$ and for any $S \subseteq [n]$ let z_S denote the monomial $\prod_{i\in S} z_i$. Recall from Section 2.2 that

$$\mathcal{E}^*(\mathcal{C}) = \prod_{i\in C^+}(1+z_i)\prod_{j\in C^-}(1-z_j) = \sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|T\cap C^-|} z_T. \tag{7}$$

From D we define a multilinear polynomial p witnessing algebraic gaps for $\mathcal{E}^*(\mathcal{C})$. (Note that the d-design is defined for the *dual* $\mathcal{C}^\dagger$ of $\mathcal{C}$, while the algebraic gaps are for $\mathcal{C}$.) We define the (multilinear) polynomial p by its coefficients: namely, for each $S \subseteq [n]$ let $\hat{p}(S) = D(z_{[n]\setminus S})$.

Clearly $\deg p = n$ since $\hat{p}([n]) = D(1) = 1$ so we focus on proving that $\deg(p\restriction\pi) \leq n-(d+1)$ for all certificates $\pi \in \mathsf{Cert}(\mathcal{E}^*(\mathcal{C}))$. This condition is equivalent to the following system of linear equations on the coefficients of $\hat{p}$: for any clause C and any subset $S \subseteq [n]$ with $S \cap \mathsf{vars}(C) = \emptyset$ and $|S| \geq n-d$ we have

$$0 = \sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|T\cap C^+|}\hat{p}(S,T). \tag{8}$$

By the definition of p, to finish the proof we must verify that

$$0 = \sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|T\cap C^+|} D(z_{[n]\setminus(S\cup T)}).$$

Letting $U = [n]\setminus(S\cup\mathsf{vars}(C))$ we can re-write this equation as

$$0 = \sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|T\cap C^+|} D(z_U z_{\mathsf{vars}(C)\setminus T}).$$

Observing that $(-1)^{|T\cap C^+|}(-1)^{|(\mathsf{vars}(C)\setminus T)\cap C^+|} = (-1)^{|C^+|}$, the linearity of D and Equation 7 implies that

$$\begin{aligned}
0 &= \sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|T\cap C^+|} D(z_U z_{\mathsf{vars}(C)\setminus T}) \\
&= D\left(\sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|T\cap C^+|} z_U z_{\mathsf{vars}(C)\setminus T}\right) \\
&= D\left(z_U\left(\sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|T\cap C^+|} z_{\mathsf{vars}(C)\setminus T}\right)\right) \\
&= D\left(z_U\left(\sum_{T\subseteq \mathsf{vars}(C)} (-1)^{|C^+|}(-1)^{|(\mathsf{vars}(C)\setminus T)\cap C^+|} z_{\mathsf{vars}(C)\setminus T}\right)\right) = (-1)^{|C^+|} D(z_U \mathcal{E}^*(C^\dagger)).
\end{aligned}$$

Since $|S| \geq n-d$ and $U = [n]\setminus(S\cup\mathsf{vars}(C^\dagger))$ we have that $|U\cup\mathsf{vars}(C^\dagger)| \leq |[n]\setminus S| \leq d$, and so $\deg(z_U\mathcal{E}^*(C^\dagger)) \leq d$, implying that $D(z_U\mathcal{E}^*(C^\dagger)) = 0$ by the design property.

□

It is natural to ask what goes wrong in the previous proof if we used the encoding $\mathcal{E}(\mathcal{C})$ instead of $\mathcal{E}^*(\mathcal{C})$. When we consider algebraic gaps this implies that the certificates of $\mathcal{E}(\mathcal{C})$ take $\{0,1\}$-values instead of $\{-1,1\}$-values. Now, instead of Equation 7 we obtain

$$\mathcal{E}(\mathcal{C}) = \prod_{i \in C^+} (1 - z_i) \prod_{j \in C^-} z_j = z_{C^-} \left(\sum_{T \subseteq \mathsf{vars}(C)} (-1)^{|T \cap C^+|} z_T \right),$$

and a problem arises with the alternating factor $(-1)^{|T \cap C^+|}$. Namely, if we consider algebraic gaps with respect to $\{0,1\}$ restrictions, we again get a system of equations on the coefficients $\hat{p}(S)$ of the polynomial p, but now without the alternation present in Equation 8. Instead, the system of equations is of the following form: for each set $S \subseteq [n]$ with $|S| \geq n - d$ we have

$$0 = \sum_{T \subseteq C^+} \hat{p}(S, T)$$

since all variables in C^- are restricted to 0 and all variables in C^+ are restricted to 1. However, over characteristic 2 this alternation is not a problem since $-1 = 1$!

Lemma 5.3. *Let* $\mathbf{F}$ *be a field of characteristic 2 and let* $\mathcal{C}$ *be an unsatisfiable* k*-CNF. Then* $\mathsf{gap}(\mathcal{E}(\mathcal{C})) = \mathsf{NS}_{\mathbf{F}}(\mathcal{C})$.

Proof. This proof is very similar to the proof of Lemma 5.2. Let D be a d-design for $\mathcal{E}(\mathcal{C}^\dagger)$. For any $S \subseteq [n]$ we let x_S denote the monomial $\prod_{i \in S} x_i$. We define the polynomial p by its coefficients in the same way as before: namely, for each $S \subseteq [n]$ let $\hat{p}(S) = D(x_{[n] \setminus S})$. Again, $\deg p = n$ since $\hat{p}([n]) = D(1) = 1$ so we focus on proving the second property. By definition, since $\mathbf{F}$ has characteristic 2 we have

$$\mathcal{E}(\mathcal{C}) = \prod_{i \in C^+} (1 - z_i) \prod_{i \in C^-} z_i = \prod_{i \in C^+} (1 + z_i) \prod_{i \in C^-} z_i = z_{C^-} \sum_{T \subseteq C^+} z_T.$$

The second condition in the definition of the gap complexity is now equivalent to the following system of linear equations on the coefficients of $\hat{p}$: for any clause $C^\dagger$ and any subset $S \subseteq [n]$ with $S \cap \mathsf{vars}(C) = \emptyset$ and $|S| \geq n - k + 1$ we have

$$0 = \sum_{T \subseteq C^-} \hat{p}(S, T).$$

By the definition of p, we must therefore verify that

$$0 = \sum_{T \subseteq C^-} D(x_{[n] \setminus S \cup T}).$$

Letting $U = [n] \setminus S \cup \mathsf{vars}(C)$ we can re-write this equation as

$$0 = \sum_{T \subseteq C^-} D(x_U x_{C^+} x_T).$$

By linearity of D this is equivalent to

$$\begin{aligned} 0 &= D\left(x_U x_{C^+} \left(\sum_{T \subseteq C^-} x_T \right) \right) \\ &= D(x_U \mathcal{E}(C^\dagger)). \end{aligned}$$

Since $|S| \geq n - d$ and $U = [n] \setminus S \cup \mathsf{vars}(C)$ we have that $|U \cup T| \leq d$, and so $\deg(x_U \mathcal{E}(C^\dagger)) \leq d$, implying that $D(x_U \mathcal{E}(C^\dagger)) = 0$ by the design property, and we have shown that $\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) \geq d+1$. To finish the proof of this direction, observe that the system $\mathcal{E}(\mathcal{C}^\dagger)$ has an $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) - 1$ design by definition, and so it follows that $\mathsf{gap}_{\mathbf{F}}(\mathcal{C}) \geq \mathsf{NS}_{\mathbf{F}}(\mathcal{C})$. □

Theorem 3.5 is an immediate corollary of Lemma 5.2 and Lemma 5.3.

6 Lifting Nullstellensatz to Algebraic Tiling

As discussed in Section 3, the upper bounds will be proven by lifting Nullstellensatz upper bounds to algebraic tiling upper bounds.

Theorem 3.3. *Let n, k be positive integers and let $\mathcal{C}$ be an unsatisfiable k-CNF with n variables and m clauses. Let $\mathbf{F}$ be a field and let g be a good gadget with rank$(g) = O(\mathsf{poly}(n))$ over $\mathbf{F}$. Then*

$$\chi_{\mathbf{F}}(f_{\mathcal{C},g}) \leq m n^{O(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))}.$$

Furthermore, suppose $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) \geq \varepsilon n$ for some constant ε independent of n. Then for any good gadget g over $\mathbf{F}$ with rank$(g) > 2^{1/\varepsilon}$,

$$\chi_{\mathbf{F}}(f_{\mathcal{C},g}) \leq m 2^{O(n)}.$$

Proof. For the sake of simplicity, let us consider the encoding $\mathcal{E}(\mathcal{C})$ and for each clause C_i let $p_i = \mathcal{E}(C_i)$. Let $q_1, q_2, \ldots, q_m$ be multilinear polynomials in $\mathbf{F}[z_1, z_2, \ldots, z_n]$ such that $\sum_{i=1}^m p_i q_i = 1$ is a minimum-degree Nullstellensatz refutation. From this we immediately have that $\sum_{i=1}^m p_i q_i \circ g^n = \mathbf{1}$, where $\mathbf{1}$ is the $\mathcal{X}^n \times \mathcal{Y}^n$ all-1s matrix. However, this is not an algebraic tiling since the matrices $p_i q_i \circ g^n$ are not necessarily embedded in the rectangles $X_{\pi,\alpha}$.

To avoid this, observe that for each polynomial p_i in $\mathcal{C}$ and each $z \in \{0,1\}^n$ we have $p_i(z) \neq 0$ if and only if z is consistent with the certificate of p_i; by extension, for all $(x, y) \in \mathcal{X}^n \times \mathcal{Y}^n$ we have that $p_i q_i \circ g^n(x, y) = 0$ unless $g^n(x, y)$ is consistent with the certificate of p_i. Therefore, letting π_i be the certificate of p_i, for each $\alpha \in \mathcal{X}^{\leq k}$ let $(p_i q_i \circ g^n) \restriction X_{\pi_i,\alpha}$ be the matrix obtained by zeroing all entries of $p_i q_i \circ g^n$ outside of $X_{\pi_i,\alpha}$. Clearly this restricted matrix is embedded within $X_{\pi_i,\alpha}$, and furthermore it is clear that the matrices $\left\{X_{\pi_i,\alpha} \mid \pi_i \in \mathsf{Cert}(\pi_i), \alpha \in \mathcal{X}^{\leq k}\right\}$ have disjoint support. Thus we can write

$$\mathbf{1} = \sum_{i=1}^m p_i q_i \circ g^n = \sum_{i=1}^m \sum_{\alpha \in \mathcal{X}^{\leq k}} (p_i q_i \circ g^n) \restriction X_{\pi_i,\alpha}.$$

Since $\mathrm{rank}(g) = \mathsf{poly}(n)$ then by Theorem 1.2

$$\mathrm{rank}_{\mathbf{F}}((p_i q_i \circ g^n) \restriction X_{\pi_i,\alpha}) \leq \mathrm{rank}_{\mathbf{F}}(p_i q_i \circ g^n) = \sum_{S : \widehat{p_i q_i}(S) \neq 0} \mathrm{rank}(g)^{|S|} \leq n^{O(\mathsf{NS}_{\mathbf{F}}(\mathcal{P}))}$$

for all $i \in [m]$ and $\alpha \in \mathcal{X}^{\leq k}$. By taking a rank-1 decomposition of the sum, this implies that

$$\chi_{\mathbf{F}}(f_{\mathcal{P},g}) \leq m \cdot |\mathcal{X}^{\leq k}| \cdot n^{O(\mathsf{NS}_{\mathbf{F}}(\mathcal{P}))} \leq m |\mathcal{X}|^{k+1} n^{O(\mathsf{NS}_{\mathbf{F}}(\mathcal{P}))} \leq m n^{O(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))}$$

since $|\mathcal{X}| = O(\mathrm{rank}(g)) = O(\mathsf{poly}(n))$ and $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) \geq k$. An analogous calculation holds if $\mathsf{NS}_{\mathbf{F}}(\mathcal{P}) \geq \varepsilon n$. □

With this we can prove Theorem 1.1 as an easy corollary.

Theorem 1.1. *Let $\mathcal{C}$ be a constant-width unsatisfiable CNF on n variables, and let $\mathbf{F}$ be any field. For any good gadget g over $\mathbf{F}$ with rank$(g) = n^2$, the lifted function $f_{\mathcal{C},g}$ satisfies*

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = n^{\Theta(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))}.$$

Further, if $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \Theta(n)$, then for any good gadget g (of sufficiently large but constant rank),

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = 2^{\Theta(n)}.$$

Proof. Let $\mathcal{C}$ be a width-k unsatisfiable CNF on n variables and let $\mathbf{F}$ be any field. In both cases, the lower bound holds by applying Theorem 3.4. The upper bound follows from Theorem 3.3 since $m \leq n^{O(k)}$. To see this, observe that in the first case $k = O(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))$ and in the second case $k = O(1)$ implies $n^{O(k)}$ is $O(\mathsf{poly}(n))$. □

References

[1] Andris Ambainis, Kaspars Balodis, Aleksandrs Belovs, Troy Lee, Miklos Santha, and Juris Smotrovs. Separations in query complexity based on pointer functions. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 800–813, 2016.

[2] Anurag Anshu, Aleksandrs Belovs, Shalev Ben-David, Mika Göös, Rahul Jain, Robin Kothari, Troy Lee, and Miklos Santha. Separations in communication complexity using cheat sheets and information complexity. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 555–564, 2016.

[3] László Babai, Anna Gál, János Kollár, Lajos Rónyai, Tibor Szabó, and Avi Wigderson. Extremal bipartite graphs and superpolynomial lower bounds for monotone span programs. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 603–611, 1996.

[4] László Babai, Anna Gál, and Avi Wigderson. Superpolynomial lower bounds for monotone span programs. *Combinatorica*, 19(3):301–319, 1999.

[5] Yakov Babichenko and Aviad Rubinstein. Communication complexity of approximate nash equilibria. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 878–889, 2017.

[6] Paul Beame, Russell Impagliazzo, Jan Krajícek, Toniann Pitassi, and Pavel Pudlák. Lower bound on hilbert's nullstellensatz and propositional proofs. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, pages 794–806, 1994.

[7] Amos Beimel. *Secure schemes for secret sharing and key distribution*. PhD thesis, Technion, 1996.

[8] Amos Beimel. *Coding and Cryptology: Third International Workshop, IWCC 2011, Qingdao, China, May 30-June 3, 2011. Proceedings*, chapter Secret-Sharing Schemes: A Survey, pages 11–46. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[9] Amos Beimel, Anna Gál, and Mike Paterson. Lower bounds for monotone span programs. *Computational Complexity*, 6(1):29–45, 1997.

[10] Amos Beimel and Enav Weinreb. Separating the power of monotone span programs over different fields. *SIAM J. Comput.*, 34(5):1196–1215, 2005.

[11] Josh Buresh-Oppenheim, Matthew Clegg, Russell Impagliazzo, and Toniann Pitassi. Homogenization and the polynomial calculus. *Computational Complexity*, 11(3-4):91–108, 2002.

[12] Samuel R. Buss. Lower bounds on nullstellensatz proofs via designs. In *Proof Complexity and Feasible Arithmetics, Proceedings of a DIMACS Workshop, New Brunswick, New Jersey, USA, April 21-24, 1996*, pages 59–72, 1996.

[13] Samuel R. Buss, Dima Grigoriev, Russell Impagliazzo, and Toniann Pitassi. Linear gaps between degrees for the polynomial calculus modulo distinct primes. *J. Comput. Syst. Sci.*, 62(2):267–289, 2001.

[14] Samuel R. Buss and Toniann Pitassi. Good degree bounds on nullstellensatz refutations of the induction principle. In *Proceedings of the Eleveth Annual IEEE Conference on Computational Complexity, Philadelphia, Pennsylvania, USA, May 24-27, 1996*, pages 233–242, 1996.

[15] Siu Man Chan and Aaron Potechin. Tight bounds for monotone switching networks via fourier analysis. *Theory of Computing*, 10:389–419, 2014.

[16] Siu On Chan, James R. Lee, Prasad Raghavendra, and David Steurer. Approximate constraint satisfaction requires large LP relaxations. *J. ACM*, 63(4):34:1–34:22, 2016.

[17] Arkadev Chattopadhyay, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Simulation theorems via pseudorandom properties. *CoRR*, abs/1704.06807, 2017.

[18] Matthew Clegg, Jeff Edmonds, and Russell Impagliazzo. Using the groebner basis algorithm to find proofs of unsatisfiability. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 174–183, 1996.

[19] Stephen A. Cook. An observation on time-storage trade off. In *Proceedings of the 5th Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1973, Austin, Texas, USA*, pages 29–33, 1973.

[20] László Csirmaz. The dealer's random bits in perfect secret sharing schemes. *Studia Sci. Math. Hungary*, 32(3-4):429–437, 1996.

[21] Susanna F. de Rezende, Jakob Nordström, and Marc Vinyals. How limited interaction hinders real communication (and what it means for proof and circuit complexity). In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 295–304, 2016.

[22] Anna Gál. A characterization of span program size and improved lower bounds for monotone span programs. *Computational Complexity*, 10(4):277–296, 2001.

[23] Anna Gál and Pavel Pudlák. A note on monotone complexity and the rank of matrices. *Inf. Process. Lett.*, 87(6):321–326, 2003.

[24] Mika Göös. Lower bounds for clique vs. independent set. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1066–1076, 2015.

[25] Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 257–266, 2015.

[26] Mika Göös and Toniann Pitassi. Communication lower bounds via critical block sensitivity. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 847–856, 2014.

[27] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. In Venkatesan Guruswami, editor, *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1077–1088. IEEE Computer Society, 2015.

[28] Mika Göös, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for BPP. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA*, pages 132–143. IEEE Computer Society, 2017.

[29] 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, 2012.

[30] Stasys Jukna. *Boolean function complexity: advances and frontiers*, volume 27. Springer Science & Business Media, 2012.

[31] Mauricio Karchmer and Avi Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM J. Discrete Math.*, 3(2):255–265, 1990.

[32] Mauricio Karchmer and Avi Wigderson. On span programs. In *Proceedings of the Eigth Annual Structure in Complexity Theory Conference, San Diego, CA, USA, May 18-21, 1993*, pages 102–111, 1993.

[33] Pravesh K. Kothari, Raghu Meka, and Prasad Raghavendra. Approximating rectangles by juntas and weakly-exponential lower bounds for LP relaxations of csps. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 590–603, 2017.

[34] 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 o of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 567–576, 2015.

[35] Igor Oliveira. *Unconditional Lower Bounds in Complexity Theory*. PhD thesis, Columbia University, 2015.

[36] Toniann Pitassi and Robert Robere. Strongly exponential lower bounds for monotone computation. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 1246–1255, 2017.

[37] Aaron Potechin. Bounds on monotone switching networks for directed connectivity. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 553–562, 2010.

[38] Pavel Pudlák and Jirí Sgall. Algebraic models of computation and interpolation for algebraic proof systems. In *Proof Complexity and Feasible Arithmetics, Proceedings of a DIMACS Workshop, New Brunswick, New Jersey, USA, April 21-24, 1996*, pages 279–296, 1996.

[39] Ran Raz and Pierre McKenzie. Separation of the monotone NC hierarchy. *Combinatorica*, 19(3):403–435, 1999.

[40] Alexander A. Razborov. Applications of matrix methods to the theory of lower bounds in computational complexity. *Combinatorica*, 10(1):81–93, 1990.

[41] Alexander A. Razborov. Lower bounds for the polynomial calculus. *Computational Complexity*, 7(4):291–324, 1998.

[42] Robert Robere, Toniann Pitassi, Benjamin Rossman, and Stephen A. Cook. Exponential lower bounds for monotone span programs. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:64, 2016.

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

[44] Avi Wigderson. $\oplus$ l/ poly = nl / poly. http://www.math.ias.edu/~avi/PUBLICATIONS/MYPAPERS/W94/proc.pdf. Accessed: 2017-09-30.

A Monotone CSP-SAT and Linear Secret Sharing Schemes

Monotone CSP-SAT. It turns out that the function $f_{\mathcal{C},g}$ has a natural interpretation as a monotone variant of the CSP-SAT problem, which we will now spell out in some detail.

Let Σ be a finite alphabet and let $H = (L \cup R, E)$ be a bipartite graph. In H, we think of the left vertices L correspond to a collection of Σ-valued variables, and the right vertices R correspond to a set of constraints over these variables. Given a constraint $C \in R$, let $\mathsf{vars}(C)$ denote the variables involved in C (or, equivalently, the neighborhood of C in the graph H).

Definition A.1. Let $H = (L \cup R, E)$ be a bipartite graph, let Σ be a finite alphabet, and let $N = \sum_{C \in R} |\Sigma|^{|\mathsf{vars}(C)|}$. The monotone function $SAT_{\Sigma,H} : \{0,1\}^N \to \{0,1\}$ is defined as follows. An input $x \in \{0,1\}^N$ defines a CSP instance $\mathcal{H}(x)$ with topology H by specifying for each constraint C in $\mathcal{H}$ a truth table $\Sigma^{\mathsf{vars}(C)} \to \{0,1\}$ of satisfying assignments to that constraint. Given an input x, $SAT_{\Sigma,H}(x) = 1$ if and only if the CSP $\mathcal{H}(x)$ is satisfiable.

For any Σ, H observe that $SAT_{\Sigma,H}$ is monotone since replacing a 0 with a 1 in the truth table of any constraint preserves the constraint's satisfying assignments, and furthermore it is clearly always computable in NP as we can guess and verify a satisfying assignment. Note that if H represents the topology of a linear-size d-CSP then $N = \Theta(n)$ if $d, |\Sigma|$ are constants.

Now, suppose that $\mathcal{C}$ is an unsatisfiable k-CSP on $\mathcal{Z}$-valued variables $z_1, z_2, \ldots, z_n$, and let $H = (L \cup R, E)$ be the constraint graph representing the topology of $\mathcal{C}$. If $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is any function then there is a natural way to convert inputs $x \in \mathcal{X}^n$ and $y \in \mathcal{Y}^n$ in the canonical search problem $\mathsf{CanSearch}(\mathcal{C}, g)$ into accepting and rejecting instances of $SAT_{\mathcal{X},H}$:

Accepting Inputs $\mathcal{U}$. Each $x \in \mathcal{X}^n$ maps to accepting input $Y(x)$ of $SAT_{\mathcal{X},H}$ for which the unique satisfying assignment to the CSP encoded by $Y(x)$ is x. Formally, for each constraint $C \in R$, the truth table $\mathcal{X}^{\mathsf{vars}(C)} \to \{0,1\}$ is entirely 0 except for a single 1 in the position $x{\restriction}\mathsf{vars}(C)$.

Rejecting Inputs $\mathcal{V}$. Each $y \in \mathcal{Y}^n$ into the rejecting input $N(y)$ of $SAT_{\mathcal{X},H}$ as follows. For each constraint $C \in R$, the truth table $t_C : \mathcal{X}^{\mathsf{vars}(C)} \to \{0,1\}$ has $t_C(\alpha) = 1$ if and only if $g^{\mathsf{vars}(C)}(\alpha, y\restriction_{\mathsf{vars}(C)}) \in \mathcal{Z}^{\mathsf{vars}(C)}$ satisfies the corresponding constraint C of the underlying CSP $\mathcal{C}$.

It is clear that the inputs $Y(x)$ are accepted by $SAT_{\mathcal{X},H}$, since the CSP encoded by $Y(x)$ is satisfied by x. To see that the inputs $N(y)$ are rejecting inputs, suppose otherwise and let $x \in \mathcal{X}^m$ be the satisfying assignment of the CSP encoded by $N(y)$. By definition of $N(y)$, it follows that for each constraint $C \in \mathcal{C}$ we have $t_C(x\restriction_{\mathsf{vars}(C)}) = 1$, which only occurs if $g^m(x, y)$ is a satisfying assignment to the CNF formula $\mathcal{C}$. This is a contradiction since $\mathcal{C}$ is unsatisfiable.

Proposition A.2. *Let $\mathcal{C}$ be an unsatisfiable CSP and let H be the bipartite graph encoding the topology of $\mathcal{C}$. The function $f_{\mathcal{C},g}$ is the same as the partial function obtained by restricting $SAT_{\mathcal{X},H}$ to $\mathcal{U}$ and $\mathcal{V}$.*

Linear Secret Sharing Schemes. Closely related to monotone span programs are *secret sharing schemes*, which are a basic cryptographic device defined as follows. We have a "dealer" who has some "secret" (say, an element k of a field $\mathbf{F}$), a set of n parties, and an upward-closed collection $\mathcal{A} \subseteq 2^{[n]}$ of subsets of the n parties called an *access structure*. A *secret sharing scheme* for $\mathcal{A}$ is a method of sharing information with the n parties such that any set of parties in $\mathcal{A}$ can reconstruct the dealer's secret, while any subset of parties not in $\mathcal{A}$ are unable to reconstruct the secret. For the sake of completeness we record the definition of secret sharing schemes here and refer the interested reader to [8] for further details.

Definition A.3. A *distribution scheme* over a domain K is a pair $\Sigma = (\Pi, \mu)$ where μ is a probability distribution over a set R and Π maps pairs in $K \times R$ to tuples $K_1 \times K_2 \times \cdots \times K_n$, where K_j is the *domain of shares* of player p_j. Given a distribution scheme Σ, a dealer distributes a secret $k \in K$ to n players as follows: first, the dealer samples a random string $r \in R$ and computes $\Pi(k, r) = (s_1, s_2, \ldots, s_n)$. Then for each $i \in [n]$, the dealer privately communicates share s_i to the i^{th} player. A distribution scheme is a *secret sharing scheme* for an access structure $\mathcal{A} \subseteq 2^{[n]}$ if it satisfies the following two properties:

Perfect Reconstruction. The secret can be reconstructed by any set of parties in the access structure, i.e. for any set of parties $A \in \mathcal{A}$ there exists a mapping $R_A : \prod_{i \in A} K_i \to K$ such that for every $k \in K$

$$\Pr[R_A(\Pi(k, r)_A) = k] = 1.$$

Perfect Privacy Every unauthorized set cannot learn anything from their shares (in the statistical sense). In other words, for any $B \notin \mathcal{A}$, for every pair of secrets $k_1, k_2 \in K$, and every vector of shares $v(s_i)_{i \in B}$ we have

$$\Pr[\Pi(k_1, r) = v] = \Pr[\Pi(k_2, r) = v].$$

The *information ratio* of a distribution scheme is $\max_{1 \le j \le n} \log |K_j| / \log |K|$, and measures the relative amount of information shared between parties. A secret sharing scheme is *linear* over a field $\mathbf{F}$ if $K = \mathbf{F}$, the random strings are field elements chosen uniformly random from $\mathbf{F}$, and the shares are vectors over $\mathbf{F}$ chosen by taking linear combinations of the secret and the random strings.

Linear secret sharing schemes are an important subclass of secret sharing schemes as many of the schemes from the literature turn out to be linear. Karchmer and Wigderson [32] proved that monotone span programs over a finite field $\mathbf{F}$ of size s for a monotone function $f : \{0,1\}^n \to \{0,1\}$ imply secret sharing schemes with information ratio s for the natural access structure associated with f (take all subsets $A \subseteq [n]$ such that $f(A) = 1$, using set-theoretic notation for boolean functions). Conversely, Beimel showed that linear secret sharing schemes imply monotone spans programs, and thus the two objects are equivalent [7].

Theorem A.4. *Let $f : \{0,1\}^n \to \{0,1\}$ be a monotone boolean function and let $\mathcal{A}_f$ denote the related access structure. For any finite field $\mathbf{F}$ there exists a monotone span program for f of size s if and only if there exists a linear secret sharing scheme for $\mathcal{A}_f$ with information ratio s.*

B Applications

Let us restate Theorem 1.1 for convenience.

Theorem 1.1. *Let $\mathcal{C}$ be a constant-width unsatisfiable CNF on n variables, and let $\mathbf{F}$ be any field. For any good gadget g over $\mathbf{F}$ with rank$(g) = n^2$, the lifted function $f_{\mathcal{C},g}$ satisfies*

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = n^{\Theta(\mathsf{NS}_{\mathbf{F}}(\mathcal{C}))}.$$

Further, if $\mathsf{NS}_{\mathbf{F}}(\mathcal{C}) = \Theta(n)$, then for any good gadget g (of sufficiently large but constant rank),

$$\mu_{\mathbf{F}}(f_{\mathcal{C},g}) = \Theta(\chi_{\mathbf{F}}(f_{\mathcal{C},g})) = 2^{\Theta(n)}.$$

Using this theorem we first characterize the complexity of computing the layered st-connectivity function by monotone span programs over all fields. Let m, n be positive integers, and let $G_{n,m}$ denote the following directed graph with $mn + 2$ vertices $V = \{v_{i,j} \mid i \in [n], j \in [m]\} \cup \{s, t\}$. We think of the vertices as being arranged in $m + 2$ layers indexed by $i = 0, 1, \ldots, m + 1$: layer 0 contains the vertex s, layer $m + 1$ contains the vertex t, and the jth layer for $j = 1, 2, \ldots, m$ contains vertices $\{v_{i,j} \mid i \in [n]\}$. Finally, for each pair of adjacent layers L_i, L_{i+1} add all edges oriented from L_i to L_{i+1}, and note that the final graph contains $mn^2 + 2n$ edges.

With this graph in mind, the *layered st-connectivity* function $\mathrm{STCONN}_{n,m}$ is defined as follows: the input is a boolean string of length $mn^2 + 2n$ describing a subgraph of the graph $G_{n,m}$ defined above, and the function outputs 1 if and only if there is a directed path from s to t. In a seminal work, Karchmer and Wigderson [31] showed that optimal monotone formulas computing $\mathrm{STCONN}_{n,m}$ have size $m^{\Omega(\log n)}$ — the upper bound follows from recursive doubling, and the lower bound was shown via communication complexity. Their lower bound was improved by Potechin [37] to hold for monotone switching networks, and by Robere, Pitassi, Rossman and Cook [42] to real monotone span programs and monotone comparator circuits. We show the same theorem holds for monotone span programs over all fields. This fact is notable as non-monotone span programs over $GF(2)$ are known to be able to compute $\mathrm{STCONN}_{n,m}$ efficiently [44].

Theorem B.1. *For all sufficiently large n and for every field $\mathbf{F}$, $\mathsf{mSP}_{\mathbf{F}}(\mathrm{STCONN}_{2n^2,n}) = n^{\Theta(\log n)}$.*

Proof. The upper bound holds since monotone span programs can simulate monotone formulas. For the other direction, Buss and Pitassi [14] show that $\mathsf{NS}_{\mathbf{F}}(IND_m) = \Omega(\log m)$ where IND_m is the unsatisfiable CNF formula

$$IND_m = z_1 \wedge (\overline{z}_1 \vee z_2) \wedge (\overline{z}_2 \vee z_3) \wedge \cdots \wedge (\overline{z}_{m-1} \vee z_m) \wedge \overline{z}_m,$$

and several previous works [26,39,42] have observed that if $g : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ is a gadget then $\mathrm{STCONN}_{|\mathcal{X}|,m}$ is a total extension of $f_{IND_m,g}$. Applying the lower bound from Theorem 1.1 completes the proof. □

Since polynomial-size monotone circuits can compute $\mathrm{STCONN}_{n,m}$, the previous theorem yields a quasipolynomial separation between $\mathsf{mSP}_{\mathbf{F}}$ and mP over all fields $\mathbf{F}$. We can improve this to a (weakly) exponential separation by considering the GEN function, defined next.

Let n be a positive integer and let $\mathcal{T} \subseteq [n]^3$ be a subset of triples of $[n]$. We say that $\mathcal{T}$ *generates* a point $w \in [n]$ if $w = 1$, or if there is a triple $(u, v, w) \in \mathcal{T}$ such that $\mathcal{T}$ generates u and v. The GEN_n problem is then defined as follows: as input, we receive a subset $\mathcal{T} \subseteq [n]^3$, encoded as a bitstring of length n^3, and must decide whether or not $\mathcal{T}$ generates the point n.

Let h be a positive integer. A *pyramid graph* of height h is the graph Δ_h on $n = \binom{h}{2}$ vertices V, which are partitioned into h sets $V_1, V_2, \ldots, V_h$ where V_i has i vertices. Ordering each V_i as $v_{i,1}, v_{i,2}, \ldots, v_{i,i}$; then for each $i = 2, 3, \ldots, h$, if $v_{i,j}$ and $v_{i,j+1}$ are adjacent vertices in V_i add two edges $(v_{i,j}, v_{i-1,j})$ and $(v_{i,j+1}, v_{i-1,j})$. A *pyramid instance* of GEN is a collection of triples $\mathcal{T}$ which is naturally isomorphic to a pyramid graph: the top point of the pyramid is n, and we assume that the point 1 is connected to each of the points $v_{1,i}$ in the first layer of the pyramid by triples $(1, 1, i)$. Define Δ_h-GEN$_n$ to be the restriction of GEN$_n$ wherein one only needs to recognize if the input generates n by a height-h pyramid instance Δ_h (necessarily $n \geq \binom{h}{2}$).

It is not hard to see that the Δ_h-GEN$_n$ problem has polynomial-size monotone circuits, and it has been used in several previous works studying the strength of circuit classes inside mP. For instance, Raz and McKenzie [39] have used the function to separate mNC_i from mNC_{i+1} for all i, and Chan and Potechin [15] used it to prove strong lower bounds against monotone switching networks.

Theorem B.2. *Let h be a positive integer and let $n = \binom{h+1}{2}$. For every field $\mathbf{F}$, $\mathsf{mSP}_{\mathbf{F}}(\Delta_h\text{-GEN}_{2n^3}) = N^{\Theta(N^\varepsilon)}$ for some constant $\varepsilon > 0$ and N is the number of input variables to the function.*

Proof. First we note that $N \leq O(n^6) \leq O(h^{12})$. The upper bound holds since there are $n^{O(h)}$ pyramid instances of GEN$_n$ of height h, and by brute force we can construct a monotone formula checking each of these with the same size. We therefore focus on the lower bound.

Consider the following unsatisfiable CNF Peb_{Δ_h}. There is one boolean variable z_v for each vertex v in Δ_h, and we have the following clauses:

1. The *target clause* $(\neg z_t)$.
2. For each source vertex $u \in R$ add the *source clause* (z_u).
3. For each internal vertex w with in-neighbours $W \subseteq V$ add the *edge clause* $\left(z_w \vee \bigvee_{v \in W} \neg z_v\right)$.

Let $g : \mathcal{X} \times \mathcal{Y} \to \mathbf{F}$ be the matrix from Theorem 1.1, and note that $|\mathcal{X}| \leq 2n^2$. It has been observed by several works [26, 39, 42] that Δ_h-GEN$_{n|\mathcal{X}|}$ is a total extension of the partial function $f_{\mathsf{Peb}_{\Delta_h}, g}$, and thus lower bounds on $\mathsf{NS}_{\mathbf{F}}(\mathsf{Peb}_{\Delta_h})$ will yield the theorem.

Buresh-Oppenheim et al. [11] show that the formula Peb_G requires Nullstellensatz degree at least the *pebbling number* of the graph G over every field. Cook [19] showed that the pebbling number of the height-h pyramid Δ_h is $\Omega(h)$. Applying the lower bound from Theorem 1.1 and using the crude bounds on N in terms of h and n yields the theorem. □

Finally, we come to the question of separating the strength of monotone span programs over different fields. Beimel and Weinreb showed that for each prime p there is a function with polynomial size monotone span programs over $GF(p)$, but all fields with characteristic different from p require monotone span programs of size $n^{\Omega(\sqrt{\log n})}$. We improve this separation to its limit: we show that for each prime p there is a function f with polynomial-size monotone span programs over fields of characteristic p, but for all fields of characteristic $q \neq p$ the function f requires monotone span programs of *strongly exponential* size (i.e $2^{\Omega(N)}$ where N is the number of input variables). The function f is also computable in NP, and thus we obtain strongly exponential lower bounds for monotone span programs over all characteristics, nearly matching the lower bounds for non-explicit functions obtained by counting arguments [30].

Theorem B.3. *For every prime p there exists a monotone boolean function f with N inputs such that f satisfies $\mathsf{mSP}_{\mathbf{F}}(f) = \mathsf{poly}(N)$ for all fields $\mathbf{F}$ of characteristic p, but for every field $\mathbf{F}'$ of characteristic $q \neq p$, $\mathsf{mSP}_{\mathbf{F}'}(f) = 2^{\Theta(N)}$. Furthermore, the function is f is computable in NP.*

Proof. Buss, Grigoriev, Impagliazzo and Pitassi [13] describe, for each positive integer n and each prime p a constant-width linear-size unsatisfiable CNF formula MOD_n^p satisfying:

1. For each field $\mathbf{F}$ of characteristic p, $\mathsf{NS}_{\mathbf{F}}(MOD_n^p) \leq O(1)$, and
2. For each field $\mathbf{F}$ of characteristic $q \neq p$, $\mathsf{NS}_{\mathbf{F}}(MOD_n^p) = \Omega(n)$.

(In fact, each of these bounds holds for the stronger polynomial calculus proof system.) Applying the upper and lower bounds, respectively, from Theorem 1.1 immediately yields the result. □

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il