Proving Algebraic Independence in Zero-Knowledge

Michael A. Forbes
Department of Computer Science, University of Illinois at Urbana-Champaign, IL, USA

Andrei Staicu
Department of Computer Science, University of Illinois at Urbana-Champaign, IL, USA

Abstract

A set of multivariate polynomials is *algebraically independent* if they exhibit no non-trivial algebraic relations, and this notion is fundamental in algebra. When these polynomials are given as algebraic circuits, deciding algebraic independence has several applications in algebraic complexity theory. Over fields of zero (or exponentially large) characteristic, this problem is known to have an efficient randomized algorithm. Over finite fields of small characteristic, a sequence of works has culminated in showing that algebraic independence admits Arthur-Merlin proofs, in particular giving the complexity bound of $\mathsf{AM} \cap \mathsf{coAM}$ ([14]).

We improve the complexity of deciding algebraic independence over finite fields by showing that it admits zero-knowledge proofs, in particular giving the upper bound of $\mathsf{NISZK} \subseteq \mathsf{AM} \cap \mathsf{coAM}$, the class of problems admitting non-interactive statistical zero-knowledge proofs. This is achieved by arguing that algebraically independent polynomials yield maps whose output distribution has high-entropy, while algebraically dependent polynomials yield maps with low-entropy. We can then reduce to the question of approximating entropy, which is a known NISZK-complete problem.

We also more generally show that *transcendence degree*, which quantifies the independence of a set of possibly dependent polynomials, can be computed in NISZK.

2012 ACM Subject Classification Theory of computation → Algebraic complexity theory; Theory of computation → Complexity classes

Keywords and phrases Algebraic Dependence, Non-Interactive Statistical Zero-Knowledge

Digital Object Identifier 10.4230/LIPIcs.ICALP.2026.93

Category Track A: Algorithms, Complexity and Games

Funding *Michael A. Forbes*: Supported by NSF CAREER award 2047310.
Andrei Staicu: Supported by NSF CAREER award 2047310.

1 Introduction

Linear algebra plays a fundamental role in mathematics, and a main question there is understanding whether a given set of linear forms $\ell_1(\overline{x}), \ldots, \ell_m(\overline{x})$ over variables $\overline{x} = (x_1, \ldots, x_n)$ are linearly dependent. Higher algebra generalizes this notion to that of *algebraic dependence*, where we ask if a given set of polynomials $f_1(\overline{x}), \ldots, f_m(\overline{x})$ are dependent, meaning there exists a non-zero annihilating polynomial P such that $P(f_1, \ldots, f_m) = 0$. If no such P exists then the polynomials $\overline{f}$ are *algebraically independent*. We can quantify beyond just dependence-vs-independence to define *transcendence degree*, which is the size of the largest subset of the polynomials f_i that are algebraically independent. Like linear independence, algebraic independence and transcendence degree play a fundamental role in understanding multivariate polynomials, for example in how the dimension of an algebraic variety is defined.

In this paper we study algebraic independence as a *computational* problem, so that given the polynomials $\overline{f}$ we seek to decide if they are independent. In formalizing this problem, one must set the encoding of the input. In the complexity-theoretic literature, it is most often

assumed that the polynomials $\overline{f}$ are given by *algebraic circuits*, which are syntactic diagrams that construct polynomials from the variables $\overline{x}$ using basic operations such as addition, multiplication, and scalars from the underlying field $\mathbb{F}$. Algebraic circuits are the most natural succinct way to present algebraic computation, see for example the survey of Shpilka and Yehudayoff [28]. This encoding is particularly well-motivated as algebraic independence then (trivially) generalizes the *polynomial identity testing (PIT)* problem, which asks whether a given algebraic circuit computes a non-zero polynomial. That is, a single polynomial f_1 is algebraically dependent iff f_1 is constant iff $f_1(\overline{x}) - f_1(0)$ is zero. As developing (deterministic) PIT algorithms is a central challenge in algebraic complexity theory, one can view the problem of deciding (succinct) algebraic independence as a far-reaching generalization of PIT.

The expressiveness of the algebraic independence testing problem leads to a number of important applications. One application is PIT itself. In particular, deterministic PIT algorithms for certain restricted classes of algebraic circuits can be solved by a general "rank bounds" methodology. One first shows a structural result, arguing that "simple and minimal" circuits from these classes that express polynomial identities necessarily have a small "rank bound". One then argues that the number of variables can (in a derandomized way) be reduced to this rank bound, at which point brute-force methodologies become comparatively efficient. This methodology was done for bounded top-fan-in depth-3 circuits, using linear independence as the notion of rank, in a series of works culminating in the work of Saxena and Seshadri [24]. This paradigm was generalized to bounded top- and middle-fan-in depth-4 circuits by [6], where the notion of rank is now that of transcendence degree. Rank bounds in this setting are much more difficult, but have been established in some settings, see Garg-Oliveira-Sengupta [10]. A key difficulty in using transcendence degree for PIT is arguing that transcendence degree is preserved in the derandomized variable reduction. This requires a reasonable mathematical criterion for computing transcendence degree; this roughly corresponds to an efficient algorithm for computing transcendence degree.

Other applications include designing extractors for the images of polynomial maps ([8]), and for computing the maximum tensor rank of a given format ([7, Chapter 20]).

Over a field of characteristic 0 (or exponentially large), it is known that one can reduce the notion of algebraic dependence to that of linear dependence. As is typical in any sort of linearization, one uses differential techniques; in particular the transcendence degree of $\overline{f}$ is equal to the (linear) rank of the Jacobian matrix of $\overline{f}$. Whether the Jacobian is full-rank then is asking if a certain determinant is zero, which can be solved using PIT techniques. This shows that algebraic independence can be solved in RP.

Unfortunately, in a field of characteristic p one has that $\frac{\partial x}{\partial x}^p = 0$, so the Jacobian criterion then fails. Further, it is known that while both linear and algebraic dependence structures form matroids over any field ([20]), it is known that there are algebraic matroids (over fields of small characteristic) that are not representable by linear dependence structures (over *any* field) ([20]). This indicates that in small characteristic linearization methods will necessarily fail to capture algebraic dependencies.

Absent the Jacobian criterion, one can still show that deciding algebraic independence can be done in PSPACE over any field. This is done using degree bounds for annihilating polynomials, alongside parallel linear algebra techniques. The gap between PSPACE and RP for general fields has thus prompted a sequence of works seeking better and better algorithms for algebraic independence. The work of [17] gave an $\mathsf{NP}^{\#\mathsf{P}}$ algorithm, and [21] gave an algorithm that ran efficiently when the "inseparable degree" was small. Guo, Saxena and Sinhababu [14] gave an $\mathsf{AM} \cap \mathsf{coAM}$ upper bound for finite fields, in particular putting this problem in the polynomial hierarchy as well as suggesting algebraic independence is not NP-hard (unless the polynomial hierarchy collapses). Incomparably, Garg and Saxena [11] gave an improved exponential-time algorithm.

1.1 Our Results

The starting point for this work is that Guo-Saxena-Sinhababu [14] seems unlikely to be unimprovable, as $\mathsf{AM} \cap \mathsf{coAM}$ is not a syntactically-defined class and as such is not known to have complete language problems (as far as we are aware). As such, it would be surprising if algebraic independence was $(\mathsf{AM} \cap \mathsf{coAM})$-complete. Furthermore, to obtain their result Guo-Saxena-Sinhababu [14] gave *two* algorithms, one for AM and one for coAM. While highly related, they used slightly different logic; it seems natural that a *single* algorithm should encompass both results.

Correspondingly, in this work we improve on the Guo-Saxena-Sinhababu [14] result, showing that algebraic independence over finite fields can be solved in *non-interactive statistical zero-knowledge*. That is, Guo-Saxena-Sinhababu [14] showed that there are Arthur-Merlin proofs certifying that independent polynomials are independent, as well as showing that there are such proofs that dependent polynomials are dependent. Arthur-Merlin proofs are interactions between an all-powerful prover (Merlin) and a resource-bounded verifier (Arthur). Babai-Moran [5] showed that while these interactions can include many rounds, one can convert them to the following type of protocol. First, Arthur sends Merlin a random string. Merlin then produces a purported proof, which Arthur then verifies. We observe here that the proof that an honest Merlin produces can be approximately simulated by a resource-bounded third party, who only knows that the polynomials are algebraically independent (and in particular knows nothing more). The approximation error between the true distribution of Merlin's proofs and the simulator's version is negligible, so this forms a *statistical zero-knowledge guarantee*: Arthur learns from Merlin that the $\overline{f}$ are algebraically independent, and nothing more. This protocol is considered to be *non-interactive* because the challenge that Arthur sends to Merlin in the first round is a completely random string independent of the input.

Formally, we have the following result.

▶ **Theorem 1.1** (Corollary 4.5). *Given a finite field $\mathbb{F}_q$ and polynomials $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ represented by size-s circuits, there is a non-interactive statistical zero-knowledge proof that $\overline{f}$ are algebraically independent. That is,* $\mathsf{AlgIndep} \in \mathsf{NISZK}$.

The above exposition immediately suggests that NISZK proofs are a subset of AM proofs, which as-is would only improve the Guo-Saxena-Sinhababu [14] result to $\mathsf{NISZK} \cap \mathsf{coAM}$. However, known complexity theoretic results show that $\mathsf{NISZK} \subseteq \mathsf{SZK}$, where SZK is the class of *statistical zero-knowledge* proofs (hence allowing more interaction). Further, it is known that $\mathsf{SZK} = \mathsf{coSZK}$ ([18]) and that $\mathsf{SZK} \subseteq \mathsf{AM}$, and as such we have that $\mathsf{NISZK} \subseteq \mathsf{AM} \cap \mathsf{coAM}$. Hence, we have a *single* algorithmic result that in particular implies the Guo-Saxena-Sinhababu [14] result. Moreover, this result exposes what we believe is the natural conclusion of the techniques of Guo-Saxena-Sinhababu [14] .

Placing algebraic independence in NISZK has potential cryptographic implications. Ostrovsky [19] showed that any problem that is hard on average, that also possesses statistical zero-knowledge proofs, will imply one-way functions (and hence all of private-key cryptography). While it seems possible that algebraic dependence in small-characteristic is easy, it also seems possible it is difficult. If so, as an algebraic problem, it would likely also be hard on average as many algebraic problems have worse-case to average-case reductions; one would then conclude that one-way functions exist. This line of thought, along with our work, suggests exploring algebraic dependence as a possible source of cryptographic hardness.

1.2 Our Techniques

Our techniques largely follow that of Guo-Saxena-Sinhababu [14] , but with a focus on incorporating zero-knowledge methods into our proofs. In particular, we view the polynomials $\overline{f}$ as a map $\overline{f} : \mathbb{F}_q^n \to \mathbb{F}_q^m$ and study the image $\mathrm{im}(\overline{f})$ of this map. Guo-Saxena-Sinhababu [14] showed that when $\overline{f}$ are independent the image is "large" (when possibly passing to a field extension $\mathbb{F}_{q^t} \supseteq \mathbb{F}_q$), while when the $\overline{f}$ are dependent the image is "small". This yields an AM protocol for algebraic independence by appealing to the AM set-size approximation protocol of Goldwasser and Sipser [13] to distinguish between large and small sets. For their coAM protocol they instead look at the pre-images of the map defined by $\overline{f}$. That is, when $\overline{f}$ are independent the image is large, and hence "most" pre-images must be small. Similarly, when the $\overline{f}$ are dependent the image is small, and hence "most" pre-images must be large. One then again appeals to the Goldwasser and Sipser [13] protocol.

We combine the ideas of the above two protocols to obtain our result. While one could construct a NISZK protocol directly, we instead use that NISZK has well-known complete (promise) problems, in particular that estimating the entropy of a distribution given by a boolean circuit is NISZK-complete. That is, one is given a boolean circuit computing a function $g : \{0,1\}^n \to \{0,1\}^m$, and one can then ask questions about the distribution of the random variable $g(\mathsf{U}_n)$ where U_n is the uniform distribution on n bits. Goldreich, Sahai and Vadhan [12] showed that there is an NISZK protocol to prove that the entropy of $g(\mathsf{U}_n)$ is "high", in comparison to when the entropy is "low"; this is a promise problem because of the gap between low- and high-entropy.

We show that for a polynomial map given by $\overline{f} : \mathbb{F}_q^n \to \mathbb{F}_q^n$, we can naturally interpret it as a boolean function $g_{\overline{f}} : \{0,1\}^m \to \{0,1\}^m$ for $m \sim n \log q$. When $\overline{f}$ are independent, then (after possibly passing to a field extension $\mathbb{F}_{q^t} \supseteq \mathbb{F}_q$) "most" pre-images of $\overline{f}$ (and hence $g_{\overline{f}}$) are small, and this implies that $g_{\overline{f}}(\mathsf{U}_m)$ has low collision probability and hence high entropy. When $\overline{f}$ are dependent, then the image of $\overline{f}$ (and hence $g_{\overline{f}}$) is "small", so that the entropy must also be small. By setting parameters carefully, one can find that there is a sufficient entropy gap in $g_{\overline{f}}(\mathsf{U}_m)$ when comparing the dependent and independent cases; completing the reduction.

However, there are various subtleties. This includes encoding issues, such as dealing with how q^n is not (generally) a power of 2. There is also that our reduction is *randomized*, due to the need to find irreducible polynomials (to construct an extension field), as well as to randomly change variables so that $\overline{f}$ is n polynomials on n variables. We then need to invoke the fact that NISZK is closed under randomized reductions, which involves some slight care because the failure probability must be negligible.

We finally note here that our techniques extend to certifying transcendence degree in NISZK.

▶ **Theorem 1.2** (Corollary 4.13)**.** *Given a finite field $\mathbb{F}_q$, $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ represented by size-s circuits, and number $r \geq 1$, there is a non-interactive statistical zero-knowledge proof that* $\text{tr-deg}(\overline{f}) \geq r$.

This is achieved by randomly reducing this problem to the case when $n = m = r$, solved above, by taking random linear combinations of the $\overline{f}$.

2 Preliminaries

Notation

We briefly here set notation for the paper, most of which is standard. For logarithms, we denote lg for $\log_2$. We will denote $\mathbb{F}$ to be a general field, and $\mathbb{F}_q$ denotes the finite field with q elements. The algebraic closure of a field $\mathbb{F}$ will be denoted as $\overline{\mathbb{F}}$. A polynomial ring in n indeterminates will be denoted by $\mathbb{F}[x_1, \ldots, x_n]$. We will write $\overline{x}$ for the tuple $(x_1, \ldots, x_n)$ (or alternatively as a column vector) when the dimensionality is clear from context. Similarly, we will write $\overline{a}$ to express a tuple from S^n for a set S. We will sometimes use the 1-norm to measure degrees of monomials, so that for $\overline{a} \in \mathbb{N}^n$, we have $|\overline{a}|_1 = \sum_{i=1}^n a_i$ (note that as $a_i \geq 0$ no absolute values are needed).

We extend the notion of the degree of a polynomial to tuples of polynomials, so that $\deg(\overline{f}) = \max_i \deg(f_i)$.

For any polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$, $\mathbb{V}_{\mathbb{F}}(f)$ will denote the set of zeros of f over $\mathbb{F}$, that is, $\mathbb{V}_{\mathbb{F}}(f) := \{\overline{\alpha} \in \mathbb{F}^n : f(\overline{\alpha}) = 0\}$. Given m polynomials $\overline{f} = (f_1, \ldots, f_m)$ over $\mathbb{F}[x_1, \ldots, x_n]$ we often abuse notation and interpret $\overline{f}$ as a map $\overline{f} : \mathbb{F}^n \to \mathbb{F}^m$ given by $\overline{f} : \overline{\alpha} \mapsto (f_1(\overline{\alpha}), \ldots, f_m(\overline{\alpha}))$. Note that we can also interpret this map over any extension field $\mathbb{K} \supseteq \mathbb{F}$.

We also denote $\overline{f}^{-1}$ as the pre-image of $\overline{f}$, so that for $\overline{\beta} \in \mathbb{F}^n$, $\overline{f}^{-1}(\overline{\beta}) = \{\overline{\alpha} \in \mathbb{F}^n : \overline{f}(\overline{\alpha}) = \overline{\beta}\}$ (which is possibly empty). We write $\overline{f}_{\mathbb{F}}^{-1}$ to emphasize the pre-image is taken over $\mathbb{F}$, but often this is omitted when clear.

We will use the following non-standard notation often, which we now highlight.

▶ **Definition 2.1.** *For polynomials $f_1, \ldots, f_m \in \mathbb{F}[\overline{x}]$, we denote their* ***degree product*** $\mathrm{D}(\overline{f})$, *as* $\mathrm{D}(\overline{f}) := \prod_{i=1}^m \deg(f_i)$.

Algebraic Dependence and Transcendence Degree

▶ **Definition 2.2.** *Polynomials $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ are* ***algebraically dependent (over $\mathbb{F}$)*** *if there exists a non-zero $g \in \mathbb{F}[y_1, \ldots, y_m]$ such that $g(\overline{f}) = 0$ in $\mathbb{F}[\overline{x}]$. They are otherwise* ***algebraically independent****. The polynomial g is said to be an* ***annihilator*** *of $\overline{f}$.*

The ***transcendence degree of $\overline{f}$ (over $\mathbb{F}$)****, denoted* $\text{tr-deg}_{\mathbb{F}}(\overline{f})$, *is the size of the largest subset $S \subseteq [m]$ such that $\overline{f}|_S = (f_i)_{i \in S}$ is algebraically independent. The field is omitted when clear from context.*

Note that the polynomials $\overline{f}$ being algebraically independent is equivalent to requiring that the map

$$\begin{aligned} \varphi : \mathbb{F}[\overline{y}] &\to \mathbb{F}[\overline{x}] \\ y_i &\mapsto f_i(\overline{x}) \ , \end{aligned}$$

is injective. The kernel of φ is precisely the set of annihilators. We note here the folklore fact that transcendence degree is preserved by moving to field extensions, which our methods will exploit.

▶ **Lemma 2.3** (e.g., Kayal [15, Claim 7.2], Beecken-Mittmann-Saxena [6, Lemma A.2])**.** *Let $\mathbb{K} \supseteq \mathbb{F}$ be an algebraic field extension of $\mathbb{F}$, and $f_1, \ldots, f_m \in \mathbb{F}[\overline{x}] \subseteq \mathbb{K}[\overline{x}]$. Then $\overline{f}$ are algebraically independent over $\mathbb{F}$ iff $\overline{f}$ are algebraically independent over $\mathbb{K}$.*

To say that $f_1, \ldots, f_m$ are algebraically dependent stipulates that there exists a non-zero annihilator g, but a priori there is no bound on the degree of g. For computational efficiency we need such a bound for annihilators, attributed to Perron, which shows that in the case of

$m-1$ variables there exists an annihilator of degree at most $\mathrm{D}(\overline{f})$. Płoski [22] has given a simple proof. The bound was sharpened by Beecken, Mittmann and Saxena [6] to reduce the exponent from m, the number of polynomials, to $\text{tr-deg}(\overline{f})$, as well as to make the number of variables arbitrary.

▶ **Lemma 2.4** (Perron's Bound ([22], [6, Corollary 6])). *Suppose $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ are algebraically dependent. Then there exists a non-zero annihilator $g \in \mathbb{F}[y_1, \ldots, y_m]$ with*

$$\deg(g) \leq (\deg(\overline{f}))^{\text{tr-deg}(\overline{f})} .$$

Testing Algebraic Independence and Transcendence Degree

We now define the computational problems of interest to this paper.

We first briefly recall how finite fields are presented, see Shoup [27] for details. For a prime power $q = p^t$, the finite field $\mathbb{F}_q$ is given by $\mathbb{F}_q = {}^{\mathbb{F}_p[z]}/_{\langle h(z)\rangle}$, where $h \in \mathbb{F}_p[z]$ is an irreducible polynomial of degree t. As such, every element of $\mathbb{F}_q$ can be represented as a degree $< t$ polynomial over $\mathbb{F}_p$, and as such they take $\Theta(t \lg p) = \Theta(\lg q)$ bits to represent in the natural encoding. Note that as primality of integers can be deterministically decided ([2]), and irreducibility of polynomials over prime fields can also be deterministically checked ([23]); hence this encoding can be efficiently and deterministically verified.

Our polynomials will be represented by algebraic circuits over the field $\mathbb{F}_q$, which is the natural succinct way to represent algebraic computation (see for example Shpilka and Yehudayoff [28]). An algebraic circuit is a labelled directed acyclic graph. The leaves are labelled with variables x_i or field elements, and internal gates are labelled with either addition or multiplication. The circuit naturally computes a polynomial from its leaves. The output of the circuit is the polynomial computed by the designated output node. The size of the circuit is the description size of the directed acyclic graph, along with the description size of the corresponding labels. When the field is $\mathbb{F}_q$ we use the above-mentioned encoding, and assume that the constant 0 is used in the circuit. It follows that the circuit size is at least $\Theta(\lg q)$ so that algorithms can be considered efficient when using $\mathbb{F}_q$ arithmetic.

▶ **Definition 2.5.** *The **algebraic independence** problem, denoted* AlgIndep*, is to decide, given algebraic circuits representing polynomials $f_1, \ldots, f_m$ over $\mathbb{F}_q$, whether the polynomials $\overline{f}$ are algebraically independent.*

Note that the input size of this problem is $\Omega(m + s)$, where s is the size of the largest algebraic circuit given and $s \geq \lg q$ by the discussion above. Thus, an algorithm that runs in $\mathsf{poly}(m, s, \lg q)$ will be efficient. We will need our reductions to be length increasing, and consequently we will need an upper bound on the input length, which is clearly $m \cdot s$.

We will also consider the more general problem of computing transcendence degree. We give here the natural decision version of this problem.

▶ **Definition 2.6.** *The **transcendence degree** problem, denoted* TrDeg*, is to decide, given algebraic circuits representing polynomials $f_1, \ldots, f_m$ over $\mathbb{F}_q$, and a number $r \in \mathbb{N}$ whether the transcendence degree of $\overline{f}$ obeys* $\text{tr-deg}(\overline{f}) \geq r$.

Non-Interactive Statistical Zero-Knowledge

Recall that the *statistical distance* of two random variables X and Y taking values on a finite set T, denoted $\Delta(\mathsf{X}, \mathsf{Y})$, is given by

$$\Delta(\mathsf{X}, \mathsf{Y}) = \sum_{\alpha} \frac{1}{2} \big| \Pr[\mathsf{X} = \alpha] - \Pr[\mathsf{Y} = \alpha] \big| .$$

The class NISZK is defined in terms of non-interactive proof systems. Recall that a promise problem is defined by a disjoint pair of sets of strings (Π_Y, Π_N) and corresponds to the problem of deciding if a string $x \in \Pi_Y \sqcup \Pi_N$ is in Π_Y or Π_N.

▶ **Definition 2.7.** *A **non-interactive statistical zero-knowledge proof system** for a promise problem, given by $\Pi = (\Pi_Y, \Pi_N)$, is defined by a triple of probabilistic machines P, V, and S where V and S run in polynomial time and P is computationally unbounded, a polynomial $r(n)$ (which gives the size of a random reference string σ) and a function $\beta(n)$ such that the conditions below hold.*

1. *(Completeness) For all $x \in \Pi_Y$, $\Pr_{\sigma \in \{0,1\}^{r(|x|)}}[V(x, \sigma, P(x, \sigma)) = 1] \geq \frac{2}{3}$.*
2. *(Soundness) For all $x \in \Pi_N$ and all provers $\widetilde{P}$, $\Pr_{\sigma \in \{0,1\}^{r(|x|)}}[V(x, \sigma, \widetilde{P}(x, \sigma)) = 1] \leq \frac{1}{3}$.*
3. *(Zero Knowledge) For all $x \in \Pi_Y$ the statistical distance between the following two distributions is at most $\beta(|x|)$:*
   a. *Choose $\sigma \in \{0,1\}^{r(|x|)}$ uniformly, sample p from $P(x, \sigma)$ and output (p, σ).*
   b. *$S(x, c)$ where the coins c for S are chosen uniformly at random.*

For any polynomial g there exists $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$, $\beta(n) < 1/g(n)$, i.e. β is a negligible function.

The class of problems that have a non-interactive statistical zero-knowledge proof system is denoted by NISZK.

Shannon Entropy

Recall the *Shannon entropy* of a random variable X, which takes values on a finite set T, is given by:

$$\mathrm{H}(\mathsf{X}) := \sum_{\alpha \in \mathrm{Supp}\, \mathsf{X}} \Pr[\mathsf{X} = \alpha] \cdot \lg(1/\Pr[\mathsf{X} = \alpha]).$$

If C is a circuit $\{0,1\}^m \to \{0,1\}^n$ we associate C with the probability distribution X_C induced on $\{0,1\}^n$ by evaluating C on the uniform distribution on $\{0,1\}^m$.

▶ **Definition 2.8.** *The promise problem **Entropy Approximation**, denoted* EA*, is defined on the set of encodings of circuits $\{0,1\}^m \to \{0,1\}^n$ outputting n bits, such that*

$$\mathsf{EA}_Y = \{(C, k) : \mathrm{H}(\mathsf{X}_C) > k + 1\}$$
$$\mathsf{EA}_N = \{(C, k) : \mathrm{H}(\mathsf{X}_C) < k - 1\} \ .$$

The main tool in our argument is the following theorem of Goldreich, Sahai, and Vadhan, showing EA is complete for the class NISZK.

▶ **Theorem 2.9** ([12, Theorem 1.3])**.** *The promise problem* EA *is complete for* NISZK*.*

Note that we do not need that EA is NISZK-hard, only that it is in NISZK.

2.1 Bounds on Polynomial Maps

In this section we look at polynomial maps $\overline{f} : \mathbb{F}_q^n \to \mathbb{F}_q^n$ and study the sizes of the images of these maps. If $\overline{f}$ are algebraically dependent, then there must exist a non-zero annihilator, so the image of $\overline{f}$ must be contained in the zero set of a polynomial, thus is small. We use a result proven by [14]; we include a proof for completeness.

If $\overline{f}$ are algebraically independent, we show that the pre-image of $\overline{f}(\overline{\alpha})$ is small for almost all $\overline{\alpha}$. A similar result was proven by [14, Lemma 7], but our proof and final result improve on theirs. Their proof uses the affine Bézout theorem in high dimensions [25], but that

argument is highly non-trivial, while our argument is considerably more elementary and direct. Additionally, we give a better bound on the number of exceptional points with large pre-image.

First we state and prove the bound for algebraically dependent polynomials. The upper bound is a consequence of Perron's bound and the Schwartz-Zippel lemma.

▶ **Lemma 2.10** ([14, Lemma 11]). *If $f_1, \ldots, f_n \in \mathbb{F}_q[x_1, \ldots, x_n]$ are algebraically dependent then:*

$$\left|\mathrm{im}(\overline{f})\right| \leq (\deg(\overline{f}) \cdot q)^{n-1}$$

Proof. Let $P \in \mathbb{F}_q[\overline{y}]$ be the minimum degree non-zero annihilator of $f_1, \ldots, f_n$, since $P(\overline{f}) \equiv 0$ in $\mathbb{F}_q[\overline{x}]$, it follows that $\mathrm{im}(\overline{f}) \subseteq \mathbb{V}_{\mathbb{F}_q}(P)$. By Lemma 2.4 $\deg P \leq (\deg(\overline{f}))^{n-1}$, so by [26] we get the upper bound $|\mathbb{V}_{\mathbb{F}_q}(P)| \leq (\deg P) q^{n-1}$ which implies $|\mathrm{im}(\overline{f})| \leq (\deg(\overline{f}) \cdot q)^{n-1}$. ◀

In the case where the polynomials are algebraically independent, we show that for a generic element of $\mathbb{F}^n$, its pre-image maps linearly to a distinct set of points that are the root of a univariate polynomial with known degree. We first show that a generic linear map $\mathbb{F}^n \to \mathbb{F}$ is injective on any finite set of points in $\mathbb{F}^n$. For brevity of notation define $\ell_{\overline{c}}(\overline{x}) = c_1 x_1 + \cdots + c_n x_n$.

▶ **Proposition 2.11.** *Let $\overline{\alpha}_1, \ldots, \overline{\alpha}_m$ be distinct points in $\mathbb{F}^n$. There is a non-zero polynomial $P \in \mathbb{F}[c_1, \ldots, c_n]$ such that if $P(\overline{\gamma}) \neq 0$ then $\ell_{\overline{\gamma}}(\overline{\alpha}_i) \neq \ell_{\overline{\gamma}}(\overline{\alpha}_j)$ for any $i \neq j$.*

Proof. Define $P \in \mathbb{F}[\overline{c}]$ by

$$P(\overline{c}) = \prod_{i<j} \left(\sum_{k=1}^{n} c_k \left((\overline{\alpha}_i)_k - (\overline{\alpha}_j)_k \right) \right).$$

Since $\overline{\alpha}_i$ are distinct points, P is the product of non-zero polynomials, thus P is non-zero. If $P(\overline{\gamma}) \neq 0$ then for all $i \neq j$, $\ell_{\overline{\gamma}}(\overline{\alpha}_i) - \ell_{\overline{\gamma}}(\overline{\alpha}_j) = \sum_{k=1}^{n} \gamma_k((\overline{\alpha}_i)_k - (\overline{\alpha}_j)_k) \neq 0$, therefore all $\ell_{\overline{\gamma}}(\overline{\alpha}_j)$ are distinct. ◀

To use the above statement, we find an annihilator of the polynomials $f_1, \ldots, f_n, \ell_{\overline{c}}(\overline{x})$ over the ring $\mathbb{F}[\overline{c}]$. In other words, this annihilator works for all choices of linear forms $\ell_{\overline{c}}(\overline{x})$. We then partially evaluate it at $\overline{f}$, and argue that outside the zero-set of a polynomial in $\mathbb{F}[\overline{x}]$, the annihilator becomes a non-zero univariate polynomial. This is enough to argue that the number of points in the pre-image at almost every point of $\mathbb{F}^n$ is small.

▶ **Lemma 2.12.** *Suppose $f_1, \ldots, f_n$ are algebraically independent, then*

$$\left| \left\{ \overline{\alpha} \in \mathbb{F}_q^n : \left| \overline{f}_{\mathbb{F}_q}^{-1}(\overline{f}(\overline{\alpha})) \right| \leq \mathrm{D}(\overline{f}) \right\} \right| \geq q^n - \left(\deg(\overline{f}) \cdot \mathrm{D}(\overline{f}) \right) q^{n-1}.$$

Proof. Define $\mathbb{F}' = \mathbb{F}_q(c_1, \ldots, c_n)$, the rational function field over variables $\overline{c}$. The collection $\{f_1, \ldots, f_n, \ell_{\overline{c}}(\overline{x})\}$ is algebraically dependent over $\mathbb{F}'[\overline{x}]$ so by [22], there exists an annihilator $Q \in \mathbb{F}'[y_1, \ldots, y_n, z]$ with $\deg_{\overline{y},z} Q \leq \mathrm{D}(\overline{f})$. After clearing denominators, we can assume Q has coefficients in $\mathbb{F}_q[\overline{c}]$. Write Q as a polynomial in z and coefficients in $\mathbb{F}_q[\overline{c}, \overline{y}]$ as

$$Q(\overline{c}, \overline{y}, z) = \sum_{i=0}^{\deg_z Q} Q_i(\overline{c}, \overline{y}) \cdot z^i,$$

where $Q_{\deg_z Q}(\overline{c}, \overline{y})$ is non-zero. Similarly, write $Q_{\deg_z Q}(\overline{c}, \overline{y})$ as a polynomial in $(\mathbb{F}_q[\overline{y}])[\overline{c}]$

$$Q_{\deg_z Q}(\overline{c}, \overline{y}) = \sum_{\overline{a}} C_{\overline{a}}(\overline{y}) \cdot \overline{c}^{\overline{a}}.$$

Thus, there exists $\overline{a}$ with $S(\overline{y}) := C_{\overline{a}}(\overline{y})$ non-zero and by the algebraic independence of $\overline{f}$ over $\mathbb{F}_q$, $S(\overline{f}) \neq 0$ and the degree bound of Q gives the upper bound $\deg_{\overline{y}} S \leq \deg_{\overline{y}} Q \leq \mathrm{D}(\overline{f})$.

For any $\overline{\alpha} \in \mathbb{F}_q^n$ with $S(\overline{f}(\overline{\alpha})) \neq 0$, suppose there exist K distinct points

$$\{\overline{\beta}_1, \ldots, \overline{\beta}_K\} \subseteq \overline{f}^{-1}_{\overline{\mathbb{F}}_q}(\overline{f}(\overline{\alpha})),$$

by Proposition 2.11, there is a non-zero polynomial $P \in \overline{\mathbb{F}}_q[\overline{c}]$ such that if $P(\overline{\gamma}) \neq 0$ then $\ell_{\overline{\gamma}}(\overline{\beta}_i)$ are all distinct. Note $S(\overline{f}(\overline{\alpha})) \neq 0$ implies $Q_{\deg_z Q}(\overline{c}, \overline{f}(\overline{\alpha})) \neq 0$, and since $\overline{\mathbb{F}}_q$ is infinite there must exist $\overline{\gamma}$ with $P(\overline{\gamma}) \neq 0$ and $Q_{\deg_z Q}(\overline{\gamma}, \overline{f}(\overline{\alpha})) \neq 0$.

In particular $Q(\overline{\gamma}, \overline{f}(\overline{\alpha}), z)$ is a non-zero polynomial in $\overline{\mathbb{F}}_q[z]$, and by definition of Q, for all $i \leq K$

$$Q(\overline{\gamma}, \overline{f}(\overline{\alpha}), \ell_{\overline{\gamma}}(\overline{\beta}_i)) = Q(\overline{\gamma}, \overline{f}(\overline{\beta}_i), \ell_{\overline{\gamma}}(\overline{\beta}_i)) = 0,$$

therefore, $K \leq \deg_z Q \leq \mathrm{D}(\overline{f})$.

We conclude that if $S(\overline{f}(\overline{\alpha})) \neq 0$, there cannot be more than $\mathrm{D}(\overline{f})$ points in $\overline{f}^{-1}_{\overline{\mathbb{F}}_q}(\overline{f}(\overline{\alpha}))$. All that is left is to bound the number of points in $\mathbb{V}_{\mathbb{F}_q}(S(\overline{f}))$ from above with [26],

$$\left|\mathbb{V}_{\mathbb{F}_q}(S(\overline{f}))\right| \leq \deg S(\overline{f}) \cdot q^{n-1} \leq \left(\deg(\overline{f}) \cdot \deg_{\overline{y}} S\right) \cdot q^{n-1} \leq \left(\deg(\overline{f}) \cdot \mathrm{D}(\overline{f})\right) q^{n-1}. \quad \blacktriangleleft$$

2.2 Generating Finite Fields

In order for the above lemmas to give non-trivial bounds, we need to create field extensions. Here we detail the randomized algorithm described by [23] and show its error probability is exponentially small. It is known how to construct large extension fields deterministically and efficiently by a result of Adleman and Lenstra [1]. However, the randomized algorithm we present here is considerably simpler and sufficient for our purposes.

Let q_0 be a power of a prime and let $q = q_0^t$. We want to generate the field $\mathbb{F}_q$ as a field extension over $\mathbb{F}_{q_0}$. Since $\mathbb{F}_q \cong \mathbb{F}_{q_0}[z]/\langle h(z)\rangle$ for any irreducible polynomial $h \in \mathbb{F}_{q_0}[z]$ of degree t, generating finite field extensions is equivalent to finding irreducible polynomials. To do this in general we need randomness. Let $I(t)$ denote the set of irreducible monic polynomials in $\mathbb{F}_{q_0}[z]$ of degree t, by [27, Theorem 19.12] we know

$$\frac{q_0^t}{2t} \leq |I(t)| \leq \frac{q_0^t}{t}.$$

[23] also gives an algorithm for testing irreducibility of a polynomial in time $\mathsf{poly}(t, \log q_0)$. This gives a randomized algorithm for finding a degree t irreducible polynomial with high probability.

▶ **Lemma 2.13.** *Given integers $N, t \geq 1$, and a representation for a field $\mathbb{F}_{q_0}$, there is a randomized algorithm running in time $\mathsf{poly}(t, N, \log q_0)$ that returns an irreducible polynomial $h \in \mathbb{F}_{q_0}[z]$ of degree t, except with probability $\delta \leq e^{-N/2t}$ in which case it returns $\perp$.*

Proof. Pick monic polynomials $h_1, \ldots, h_N$ of degree t uniformly and independently by picking the $N \cdot t$ coefficients uniformly from $\mathbb{F}_{q_0}$. Test each h_i for irreducibility using the algorithm given in [23]. If some h_i is irreducible, halt and output h_i. If all h_i are reducible output $\perp$.

Correctness. It is obvious that if any polynomial is returned it must be irreducible.

Success Probability. Let E be the event all h_i are reducible and E_i the event h_i is reducible, we know $\Pr[E_i] \leq 1 - 1/2t$, which implies

$$\Pr[E] = \Pr\left[\bigcap_{i=1}^{N} E_i\right] = \prod_{i=1}^{N} \Pr[E_i] \leq \left(1 - \frac{1}{2t}\right)^N \leq e^{-N/2t}.$$

The probability some h_i is irreducible is at least $1 - e^{-N/2t}$, and by correctness of the irreducibility test of [23] it will be output. This gives the desired error probability.

Complexity. It takes time $\mathsf{poly}(N, t, \log q_0)$ to write down $h_1, \ldots, h_N$, and testing if h_i is irreducible by [23] takes time $\mathsf{poly}(t, \log q_0)$, thus the algorithm runs in $\mathsf{poly}(N, t, \log q_0)$. ◀

2.3 Randomized Reductions for NISZK

The reductions we develop in this paper will use randomness in several ways. In particular, we will take random linear combinations of polynomials, random changes of bases, and randomness for constructing finite fields. We note randomness for constructing finite fields is not strictly essential due to [1]. Common definitions of randomized reductions allow for a constant or inverse polynomial error probability, NISZK is not currently known to be closed under such reductions, but is known to be closed under randomized reductions with negligible error [3].

▶ **Definition 2.14.** *A promise problem $\Pi = (\Pi_Y, \Pi_N)$ is $\leq_m^{BPP}$-reducible to a promise problem $\Pi' = (\Pi'_Y, \Pi'_N)$ with error probability $\delta < 1/2$ if there is a polynomial p and a deterministic Turing machine M running in time p such that,*

- *$x \in \Pi_Y$ implies that $\Pr_{r \in \{0,1\}^{p(|x|)}}[M(x,r) \in \Pi'_Y] \geq 1 - \delta$,*
- *$x \in \Pi_N$ implies that $\Pr_{r \in \{0,1\}^{p(|x|)}}[M(x,r) \in \Pi'_N] \geq 1 - \delta$.*

*We write $\Pi \leq_m^{BPP} \Pi'$. If $|M(x,r)| \geq |x|$ for all r then we say the reduction is **length-increasing**.*

The issue with constant error $\leq_m^{BPP}$-reductions is that we only have a zero-knowledge on YES instances of Π'. If we use the natural zero knowledge proof system for Π, created by the "pullback" of the NISZK protocol for Π' via the reduction M, a constant fraction of $M(x, \cdot)$ for some $x \in \Pi_Y$ will be sent outside Π'_Y. A valid proof system for Π' could reveal information to the verifier on these inputs, in which case this proof system for Π is not zero knowledge. To avoid this issue, we require that the error probability of a randomized reduction is exponentially small in the input length.

Furthermore, using this proof system for Π the statistical distance between the output of the prover and simulator on an input x is $1/|M(x,r)|^{\omega(1)}$. To ensure that this value is indeed $1/|x|^{\omega(1)}$, we require that the reduction M is length-increasing. We note however that most natural reductions, in particular all of our reductions, are length-increasing.

▶ Remark 2.15. If there exists $\epsilon > 0$ such that for all x and r, $|M(x,r)| \geq |x|^\epsilon$, then we say the reduction is *honest*. It is clear that length-increasing reductions are honest. [3, Theorem 14] shows that NISZK is closed under honest $\leq_m^{BPP}$ reductions with exponentially small error probability by passing through Kolmogorov complexity, we will only need the following weaker statement, which we prove in the full version.

▶ **Lemma 2.16.** NISZK *is closed under $\leq_m^{BPP}$ length-increasing reductions that have error probability $\delta \leq 2^{-|x|}$.*

3 Algebraic Independence of n Polynomials in NISZK

We first focus on the case where the number of polynomials equals the number of variables. We show that the problem corresponding to instances of AlgIndep with the additional restriction that the number of polynomials is equal to the number of variables is in NISZK. We denote this problem by $\mathsf{AlgIndep}_{=}$.

▶ **Definition 3.1.** *The **algebraic independence with equal variables and polynomials** problem, denoted* $\mathsf{AlgIndep}_{=}$*, is to decide, given algebraic circuits representing polynomials* $f_1, \ldots, f_n$ *over* $\mathbb{F}_q$*, whether* $\overline{f}$ *are algebraically independent.*

We show that deciding if n polynomials in n variables are algebraically independent is in NISZK by reducing to EA. Our goal is to use the difference in image sizes shown previously to design a boolean circuit C from polynomials $f_1, \ldots, f_n \in \mathbb{F}_{q_0}[x_1, \ldots, x_n]$ where the entropy differs significantly depending on if the polynomials are algebraically dependent or algebraically independent.

3.1 Circuit Construction

Notice that EA takes in an encoding of a boolean circuit. Therefore, to describe a reduction to EA from $\mathsf{AlgIndep}_{=}$, we need to turn the algebraic circuits into boolean circuits and evaluate at an approximately uniform point in $\mathbb{F}_q^n$. As the characteristic of $\mathbb{F}_q$ may not be exactly 2, we design a map $\{0,1\}^L \to \mathbb{F}_q$ where $2^L \gg q$, and take any bit-string to its value modulo q. This turns out to output a nearly uniform element of $\mathbb{F}_q$.

▶ **Definition 3.2.** *Given some enumeration of* $\mathbb{F}_q$ *and an integer* L*, define* $\mathrm{Dec}_{q,L} : \{0,1\}^L \to \mathbb{F}_q$ *by taking* $\overline{b}$ *to the element* $\alpha_k \in \mathbb{F}_q$ *where*

$$k = \left(\sum_{i=1}^{L} b_i 2^{i-1} \right) \mod q.$$

For any integer n*, let* $\mathrm{Dec}_{q,L}^n : \{0,1\}^{nL} \to \mathbb{F}_q^n$ *by* $(\overline{b}_1, \ldots, \overline{b}_n) \mapsto (\mathrm{Dec}_{q,L}(\overline{b}_1), \ldots, \mathrm{Dec}_{q,L}(\overline{b}_n))$.

With this definition we can describe the construction of the circuit from $\mathsf{AlgIndep}_{=}$ to EA.

▶ **Construction 3.3.** *Given* $f_1, \ldots, f_n$ *as algebraic circuits over* $\mathbb{F}_{q_0}[x_1, \ldots, x_n]$*,* $h(z) \in \mathbb{F}_{q_0}[z]$ *irreducible of degree* t*, and an integer* L*, let* $C_{f,h,L}$ *be a circuit on* nL *bits, outputting* $n\lceil \lg q_0^t \rceil$ *bits, representing the function* $F_{f,h,L}(\overline{b}) = \overline{f}(\mathrm{Dec}_{q_0^t,L}^n(\overline{b})) : \{0,1\}^{nL} \to \mathbb{F}_q^n$ *represented as* $\mathbb{F}_{q_0}[z]/\langle h \rangle$.

We will show now that the algorithm described above is computable in polynomial time. This implies that we can output the encoding of the circuit which computes this function in polynomial time as well.

▶ **Proposition 3.4.** *Given* $f_1, \ldots, f_n$ *as algebraic circuits over* $\mathbb{F}_{q_0}[x_1, \ldots, x_n]$ *each of size at most* s*, an irreducible polynomial* $h \in \mathbb{F}_{q_0}[z]$ *of degree* t*, and an integer* L*, the encoding for* $C_{f,h,L}$ *can be computed in time* $\mathsf{poly}(n, L, t, s, \lg q_0)$.

Proof. It suffices to show that $F_{f,h,L}(\overline{b})$ can be computed in time $\mathsf{poly}(n, L, t, \lg q_0)$. For any L-bit integer a, the value $a \mod q_0^t$ can be computed in time $\mathsf{poly}(L)$, so the corresponding element $\alpha \in \mathbb{F}_q$ can be found in $\mathsf{poly}(t, \lg q_0)$, and decoding all n integers can be done in $\mathsf{poly}(n, t, \lg q_0)$

Addition and multiplication in $\mathbb{F}_q$ represented as $\mathbb{F}_{q_0}[z]/\langle h(z)\rangle$ can be done in $\mathsf{poly}(t)$ operations over $\mathbb{F}_{q_0}[z]$, addition and multiplication in $\mathbb{F}_{q_0}$ can be done in $\mathsf{poly}(\lg q_0)$ operations, so addition and multiplication can be done in $\mathsf{poly}(t, \lg q_0)$. This implies, given the algebraic circuit of f_i, we can compute $f_i(\overline{\alpha})$ on any $\alpha \in \mathbb{F}_q^n$ in $\mathsf{poly}(\lg q_0, t, s)$ time. Hence we can compute $F_{f,h,L}$ in time $\mathsf{poly}(n, L, t, s, \lg q_0)$ as desired. ◀

3.2 Entropy of the Circuit

Throughout this section let U_m be the uniform distribution on $\{0,1\}^m$. Recall that $F_{f,h,L}$ is defined by an irreducible polynomial $h \in \mathbb{F}_{q_0}[z]$ of degree t, so we will think of $q = q_0^t$ in the section.

Here we show that if the input polynomials are algebraically independent then the entropy of the random variable $\mathsf{X} := F_{f,h,L}(\mathsf{U}_{nL})$ can be made large. If the polynomials are algebraically dependent, then the entropy is necessarily smaller. We first deal with the case where the polynomials are algebraically dependent, this is immediate from Lemma 2.10.

▶ **Proposition 3.5.** *If $f_1, \ldots, f_n$ are algebraically dependent, then* $\mathrm{H}(\mathsf{X}) \leq (n-1)\lg(\deg(\overline{f})\cdot q)$.

Proof. By construction $\operatorname{Supp}\mathsf{X} \subseteq \operatorname{im}\overline{f}$ and by Lemma 2.10, $|\operatorname{im}\overline{f}| \leq \deg(\overline{f})^{n-1}\cdot q^{n-1}$, therefore $\mathrm{H}(\mathsf{X}) \leq \lg|\operatorname{Supp}\mathsf{X}| \leq (n-1)\lg(\deg(\overline{f})\cdot q)$. ◀

In the case the polynomials are algebraically independent, in order to give a lower bound on $\mathrm{H}(\mathsf{X})$ we need to give tight upper and lower bounds of $\Pr[\mathsf{X} = \overline{\beta}]$ for $\overline{\beta} \in \mathbb{F}_q^n$. First, we need to argue that the distribution arising from evaluating the decoding map on the uniform distribution on $\{0,1\}^{nL}$ is approximately uniform on $\mathbb{F}_q^n$.

▶ **Proposition 3.6.** *For any $\overline{\alpha} \in \mathbb{F}_q^n$*

$$\left(\frac{1-q/2^L}{q}\right)^n \leq \Pr[\mathrm{Dec}_{q,L}^n(\mathsf{U}_{nL}) = \overline{\alpha}] \leq \left(\frac{1+q/2^L}{q}\right)^n.$$

Proof. Each coordinate of $\mathrm{Dec}_{q,L}^n$ is independent, so it suffices to show for any $\alpha \in \mathbb{F}_q$ that $(1-q/2^L)/q \leq \Pr[\mathrm{Dec}_{q,L}(\mathsf{U}_L) = \alpha] \leq (1+q/2^L)/q$. If we enumerate $\mathbb{F}_q$, by the definition of $\mathrm{Dec}_{q,L}$

$$\Pr[\mathrm{Dec}_{q,L}(\mathsf{U}_L) = \alpha_i] = \Pr[\mathsf{U}_L = i \mod q].$$

Since $\Pr[\mathsf{U}_L = i \mod q] = |\{k \in \mathbb{N} \mid kq + i \leq 2^L - 1\}|/2^L$, we have:

$$\Pr[U_L = i \mod q] = \begin{cases} \lfloor 2^L/q\rfloor/2^L & \text{if } i \geq 2^L \mod q \\ \lceil 2^L/q\rceil/2^L & \text{otherwise .} \end{cases}$$

The claim follows from $\lfloor 2^L/q\rfloor/2^L \geq (1-q/2^L)/q$ and $\lceil 2^L/q\rceil/2^L \leq (1+q/2^L)/q$. ◀

As X has support only on the image of $\overline{f}$, we can split $\operatorname{Supp}\mathsf{X}$ into points with small pre-image and large pre-image. By Lemma 2.12, when $\overline{f}$ are algebraically independent there are sufficiently many points in the image of $\overline{f}$ with small pre-image that their contribution to entropy alone grows like $\Theta(n\lg q)$ for the appropriate choice of L.

▶ **Proposition 3.7.** *If $f_1, \ldots, f_n$ are algebraically independent and $q < 2^L$ then*

$$\mathrm{H}(\mathsf{X}) > \left(1 - \frac{\deg\overline{f}\cdot \mathrm{D}(\overline{f})}{q} - \frac{nq}{2^L}\right) n\lg q - \left(1 + \frac{n\deg\overline{f}\cdot\mathrm{D}(\overline{f})}{2^L}\right)\left(n + \lg \mathrm{D}(\overline{f})\right).$$

Proof. Let $A \subseteq \operatorname{im} \overline{f}$ be the set of points $\overline{\beta} \in \operatorname{im} \overline{f}$ such that $|\overline{f}^{-1}(\overline{\beta})| \leq \mathrm{D}(\overline{f})$. From Proposition 3.6, we know that for $\overline{\beta} \in A$

$$\Pr[\mathsf{X} = \overline{\beta}] = \sum_{\overline{\alpha} \in \overline{f}^{-1}(\overline{\beta})} \Pr[\mathrm{Dec}^n_{q,L}(\mathsf{U}_{nL}) = \overline{\alpha}] \leq \mathrm{D}(\overline{f}) \left(\frac{1 + q/2^L}{q} \right)^n .$$

Now we can look at the contribution of X restricted to A to $\mathrm{H}(\mathsf{X})$

$$\begin{aligned} \mathrm{H}(\mathsf{X}) &\geq \sum_{\overline{\beta} \in A} \Pr[\mathsf{X} = \overline{\beta}] \lg \left(\frac{1}{\Pr[\mathsf{X} = \overline{\beta}]} \right) \geq (\Pr[\mathsf{X} \in A]) \cdot \lg \left(\frac{q^n}{\mathrm{D}(\overline{f}) \cdot (1 + q/2^L)^n} \right) \\ &\geq \left(\sum_{\overline{\alpha} \in \overline{f}^{-1}(A)} \Pr[\mathrm{Dec}^n_{q,L}(\mathsf{U}_{nL}) = \overline{\alpha}] \right) \cdot \lg \left(\frac{q^n}{\mathrm{D}(\overline{f}) \cdot (1 + q/2^L)^n} \right) . \end{aligned}$$

By Lemma 2.12, we know that $|\overline{f}^{-1}(A)| \geq q^n - (\deg \overline{f} \cdot \mathrm{D}(\overline{f}))q^{n-1}$, by the lower bound of Proposition 3.6 we get

$$\begin{aligned} \mathrm{H}(\mathsf{X}) &\geq \left(q^n - (\deg \overline{f} \cdot \mathrm{D}(\overline{f}))q^{n-1}\right) \cdot \left(\frac{1 - q/2^L}{q} \right)^n \cdot \lg \left(\frac{q^n}{\mathrm{D}(\overline{f}) \cdot (1 + q/2^L)^n} \right) \\ &\geq \left(1 - \frac{\deg \overline{f} \cdot \mathrm{D}(\overline{f})}{q} \right) \left(1 - \frac{nq}{2^L} \right) \left(n \lg q - n \lg(1 + q/2^L) - \lg \mathrm{D}(\overline{f}) \right) . \end{aligned}$$

The final inequality follows from Bernoulli's inequality since $q/2^L < 1$. Additionally, $q/2^L < 1$ implies $\lg(1 + q/2^L) < 1$, so after dropping all positive terms from the expression, apart from $n \lg q$, we get

$$\mathrm{H}(\mathsf{X}) > \left(1 - \frac{\deg \overline{f} \cdot \mathrm{D}(\overline{f})}{q} - \frac{nq}{2^L} \right) n \lg q - \left(1 + \frac{n \deg \overline{f} \cdot \mathrm{D}(\overline{f})}{2^L} \right) \left(n + \lg \mathrm{D}(\overline{f}) \right) ,$$

which proves the claim. ◀

3.3 Reduction to EA

Recall $\mathsf{X} = F_{f,h,L}(\mathsf{U}_{nL})$ is parametrized by $f_1, \ldots, f_n \in \mathbb{F}[\overline{x}]$, an irreducible polynomial $h \in \mathbb{F}_{q_0}[z]$ of degree t, and an integer L.

From Proposition 3.4, we know we can output the encoding for Construction 3.3 efficiently in terms of t, L, s, n, and $\lg q_0$, and if the reduction runs polynomial in s, $\lg q_0$, and n, then it is polynomial in the input length to $\mathsf{AlgIndep}_=$. Therefore, all that is left to check is that there are values for t and L that are polynomial in s, $\lg q_0$, and n such that the entropy lower bound of X when $\overline{f}$ are independent is larger than the entropy upper bound of X when $\overline{f}$ are dependent.

▶ **Proposition 3.8.** *Suppose $f_1, \ldots, f_n$ are algebraically independent, $L > 2 \lg n + t \lg q_0 + 2$, and $t \lg q_0 > 6(n+1) \lg(\deg \overline{f}) + 4n + 8$ then*

$$\mathrm{H}(\mathsf{X}) > (n-1) \lg(\deg \overline{f} \cdot q) + 4.$$

Proof. Since $L > t \lg q_0$, we can apply Proposition 3.7 to get the lower bound

$$\mathrm{H}(\mathsf{X}) > \left(1 - \frac{\deg \overline{f} \cdot \mathrm{D}(\overline{f})}{q_0^t} - \frac{nq_0^t}{2^L} \right) n \lg q_0^t - \left(1 + \frac{n \deg \overline{f} \cdot \mathrm{D}(\overline{f})}{2^L} \right) \left(n + \lg \mathrm{D}(\overline{f}) \right) .$$

The bound $L > 2\lg n + t\lg q_0 + 2$ implies $2^L > 4n^2 q_0^t$, equivalently $nq_0^t/2^L < 1/4n$. Similarly, using the observation $(\deg \overline{f})^{n+1} \geq \deg(\overline{f}) \cdot \mathrm{D}(\overline{f})$ we see

$$t\lg q_0 > 6(n+1)\lg(\deg \overline{f}) + 4n + 8 \geq \lg(\deg(\overline{f}) \cdot \mathrm{D}(\overline{f})) + \lg(4n) = \lg(4n \deg(\overline{f}) \cdot \mathrm{D}(\overline{f})).$$

Therefore, $q_0^t > 4n \deg(\overline{f}) \cdot \mathrm{D}(\overline{f})$, which implies both $\deg(\overline{f})\,\mathrm{D}(\overline{f})/q_0^t + nq_0^t/2^L \leq 1/2n$ and $n \deg(\overline{f})\,\mathrm{D}(\overline{f})/2^L \leq 1/16n^2$ and simplifies the lower bound on $\mathrm{H}(\mathsf{X})$ to

$$\begin{aligned}\mathrm{H}(\mathsf{X}) &> \left(1 - \frac{1}{2n}\right) n\lg q_0^t - \left(1 + \frac{1}{16n^2}\right)\left(n + \lg \mathrm{D}(\overline{f})\right) \\ &> n\lg q_0^t - \frac{1}{2}\lg q_0^t - 2n - 2\lg \mathrm{D}(\overline{f}).\end{aligned}$$

Re-using the lower bound on $t\lg q_0$ and $(\deg \overline{f})^{n+1} \geq \mathrm{D}(\overline{f})$ we find that

$$\frac{1}{2}\lg q_0^t > 3(n+1)\lg(\deg \overline{f}) + 2n + 4 > (n-1)\lg(\deg \overline{f}) + 2\lg(\mathrm{D}(\overline{f})) + 2n + 4$$

and thus

$$\mathrm{H}(\mathsf{X}) > (n-1)\lg q_0^t + \frac{1}{2}\lg q_0^t - 2n - 2\lg \mathrm{D}(\overline{f}) > (n-1)\lg(\deg \overline{f} \cdot q) + 4. \qquad \blacktriangleleft$$

With this technical statement, we can construct the encoding for the circuit C_f that maps to a YES instance of EA if $f_1, \ldots, f_n$ are algebraically independent and a NO instance of EA if $f_1, \ldots, f_n$ are algebraically dependent, in time polynomial in the input length. In our reduction we use the following fact that shows that encodings of boolean circuits can be padded.

▶ **Lemma 3.9.** *Given a boolean circuit C of size s computing $f : \{0,1\}^m \to \{0,1\}^n$ and N in unary, there is a polynomial time algorithm that outputs a boolean circuit for f of size $\Theta(s + N)$.*

We proceed to prove the main theorem.

▶ **Theorem 3.10.** *There is a length-increasing randomized reduction from $\mathsf{AlgIndep}_=$ to EA with error probability $\delta \leq 2^{-\ell}$ on inputs of size ℓ, that is, $\mathsf{AlgIndep}_= \leq_m^{\mathrm{BPP}} \mathsf{EA}$.*

Proof. We are given $f_1, \ldots, f_n \in \mathbb{F}_{q_0}[x_1, \ldots, x_n]$ as algebraic circuits each of size at most s, suppose the input size is ℓ. We will produce a boolean circuit C such that if the $\overline{f}$ are independent C has high entropy and if $\overline{f}$ are dependent C has low entropy. As the reduction needs to be length-increasing, we require an upper bound on the length of the input, which can be taken to be $n \cdot s$.

We will evaluate the polynomials over a field extension of $\mathbb{F}_{q_0}$, which is encoded as bits. To use the tools above, we need to construct a large field extension of $\mathbb{F}_{q_0}$ of degree t, which we denote by $\mathbb{F}_q$. L is the encoding length of each $\mathbb{F}_q$, we need to pick L large to ensure that uniform on U_L is approximately uniform on $\mathbb{F}_q$.

We want to use Rabin's algorithm (Lemma 2.13), which gives an error probability of $\exp(-N/2t)$. For this to be exponentially small in the input size we set $N = \lceil 2t(n \cdot s) \rceil$. Pick $t = \lceil 6(n+1)\lg(\deg \overline{f}) + 4n + 9 \rceil$ and $L = \lceil 2\lg n + t\lg q_0 + 3 \rceil$ to satisfy the conditions for Proposition 3.8.

Run Rabin's algorithm with N and t. Suppose it outputs an irreducible polynomial h of degree t. Compute the encoding for the circuit C from Construction 3.3 with $f_1, \ldots, f_n$, L, and h as input. We can pad C so that it is larger than the input size. Compute the entropy threshold k, which is $k = \lceil (n-1)\lg(\deg \overline{f} \cdot q_0^t) \rceil + 2$ and output (C, k).

If the algorithm does not output an irreducible polynomial, then output a NO instance of EA, we can pad to ensure the encoding has size at least $n \cdot s$ by Lemma 3.9.

Length-increasing. The reduction is always length increasing as we pad the output appropriately.

Complexity. By Lemma 2.13, we can find h in $\mathsf{poly}(n, s, \lg q_0)$ time by the definition of t and N. By Proposition 3.4, the encoding for C can be computed in time $\mathsf{poly}(n, s, t, \lg q_0, L)$. The chosen values for t and L are polynomial in s, t, and $\lg q_0$, thus the encoding for C can be computed in time $\mathsf{poly}(n, s, \lg q_0)$.

Completeness. We need to show that if $f_1, \ldots, f_n$ are algebraically independent that C encodes a distribution with entropy strictly larger than $k+1$ with high probability.

Suppose that Rabin's algorithm finds an irreducible polynomial h of degree t. Notice that C encodes the distribution $\mathsf{X} = \overline{f}(\mathrm{Dec}^n_{q,L}(\mathsf{U}_{nL}))$ and t and L satisfy the assumptions for Proposition 3.8, which implies

$$\mathrm{H}(\mathsf{X}) > (n-1)\lg(\deg \overline{f} \cdot q_0^t) + 4 \geq \lceil (n-1)\lg(\deg \overline{f} \cdot q_0^t) \rceil + 3 = k+1,$$

and thus (C, k) is in EA_{Y}. Therefore, the probability that (C, k) is not in EA_{Y} is bounded from above by the probability the algorithm fails to find an irreducible polynomial. By Lemma 2.13 the error probability is $\exp(-n \cdot s) < 2^{-\ell}$.

Soundness. We need to show that if $f_1, \ldots, f_n$ are algebraically dependent then C encodes a distribution with entropy strictly smaller than $k-1$.

Suppose Rabin's algorithm finds an irreducible polynomial h. As above, C encodes the distribution $\mathsf{X} = \overline{f}(\mathrm{Dec}^n_{q,L}(\mathsf{U}_{nL}))$, and by Proposition 3.5 we know that

$$\mathrm{H}(\mathsf{X}) < (n-1)\lg(\deg \overline{f} \cdot q_0^t) \leq \lceil (n-1)\lg(\deg \overline{f} \cdot q_0^t) \rceil + 1 = k-1,$$

therefore $(C, k) \in \mathsf{EA}_{\mathrm{N}}$. If Rabin's algorithm fails to find an irreducible polynomial, we output a NO instance of EA. Hence, (C, k) is always in EA_{N}. ◀

Therefore, by Lemma 2.16, we see that $\mathsf{AlgIndep}_=$ is indeed in NISZK as desired.

▶ **Corollary 3.11.** $\mathsf{AlgIndep}_= \in \mathsf{NISZK}$.

4 Changing Parameters in Algebraic Dependence Testing

Previously we showed that testing if n polynomials are algebraically independent in n variables is in NISZK. We now show that the general problem can be reduced to this case.

We prove here two facts. First, composing a random affine map from m variables to n variables with m algebraically independent polynomials on n variables keeps the polynomials algebraically independent, giving the reduction from $\mathsf{AlgIndep}$ to $\mathsf{AlgIndep}_=$. We then prove that r random linear combinations of m polynomials are algebraically independent if their transcendence degree was at least r, giving the reduction from TrDeg to $\mathsf{AlgIndep}$.

4.1 Changing the Number of Variables

In this section we show how to reduce the number of variables from n to the number of polynomials m while maintaining algebraic independence. Similar results were achieved by [9, Theorem 5.2.4] using more sophisticated algebraic geometry, our techniques are more elementary.

The following statement follows from the elementary proof of [6, Lemma 16]: that there *exists* an affine map from $\mathbb{F}[x_1, \ldots, x_n]$ to $\mathbb{F}[y_1, \ldots, y_m]$ that does not affect transcendence degree.

▶ **Lemma 4.1** ([6, Lemma 16]). *Let $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ be algebraically independent polynomials and let $\overline{y} = (y_1, \ldots, y_m)$. If $|\mathbb{F}| > m \cdot \deg \overline{f} \cdot \mathrm{D}(\overline{f})$ then there exists $A \in \mathbb{F}^{n \times m}$ and $\overline{\beta} \in \mathbb{F}^n$ such that $f_1(A\overline{y} + \overline{\beta}), \ldots, f_m(A\overline{y} + \overline{\beta})$ are algebraically independent over $\mathbb{F}$.*

This lemma by itself is insufficient for our purposes, because the construction they give uses the computation of transcendence degree. Instead, we will use this lemma to show that *generic* affine transformations also preserve algebraic independence. We want to find a criterion for the entries of a generic affine linear transformation $\overline{x} \mapsto A\overline{y} + \overline{\beta}$ that governs whether the transformation $\overline{f}(\overline{x}) \mapsto \overline{f}(A\overline{y} + \overline{\beta})$ preserves the algebraic independence of $\overline{f}$.

Specifically, we see that the bad choices for an affine linear transformation are contained in a zero locus of a polynomial. By computing the degree of this polynomial, we get an explicit bound on the probability of a good variable reduction by Schwartz-Zippel. The proof of the following statement is deferred to the full version.

▶ **Lemma 4.2.** *Let $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ be algebraically independent polynomials, $Z = (z_{1,1}, \ldots, z_{n,m})$ an $n \times m$ matrix of indeterminates, $\overline{w} = (w_1, \ldots, w_n)$, and $\overline{y} = (y_1, \ldots, y_m)$, then there exists $P \in \mathbb{F}[Z, \overline{w}]$ with $\deg P \leq (\deg \overline{f})^m (m + \deg(\overline{f})^{m-1})^m$ such that if $P(A, \overline{\beta}) \neq 0$ for $A \in \mathbb{F}^{n \times m}$ and $\overline{\beta} \in \mathbb{F}^n$ then $f_1(A\overline{y} + \overline{\beta}), \ldots, f_m(A\overline{y} + \overline{\beta})$ are algebraically independent over $\mathbb{F}$.*

Here we show that testing algebraic independence of polynomials is in NISZK by reducing to the case where we have an equal number of polynomials and variables. The above statement allows us to ensure that if the field is large enough, then with high probability the reduction of variables does not decrease the transcendence degree of the output. In our reduction we use the following fact that shows that encodings of algebraic circuits can be padded.

▶ **Lemma 4.3.** *Given an algebraic circuit computing $f \in \mathbb{F}[\overline{x}]$ of size s and N in unary, there is a polynomial time algorithm that outputs an algebraic circuit for f of size $\Theta(s + N)$.*

We now proceed to prove the main theorem.

▶ **Theorem 4.4.** *There is a length-increasing randomized reduction with error probability $\delta \leq 2^{-\ell}$ on inputs of size ℓ from* AlgIndep *to* AlgIndep$_=$*, that is,* AlgIndep $\leq_m^{\mathrm{BPP}}$ AlgIndep$_=$.

Proof. We are given $f_1, \ldots, f_m \in \mathbb{F}_{q_0}[x_1, \ldots, x_n]$ as algebraic circuits each of size at most s. We will produce $g_1, \ldots, g_m$ over a field extension $\mathbb{F}_q[y_1, \ldots, y_m]$. As the reduction needs to be length increasing, we require an upper bound on an input of length ℓ, which can be taken to be $m \cdot s$.

If $m > n$ we output a circuit computing the zero polynomial, padding to ensure the output is larger than the input. Trivially $\overline{f}$ are algebraically dependent, so the reduction has zero error probability. It is clearly length-increasing and runs in polynomial time. For the remainder of the proof we assume $m \leq n$.

We will pass to a field extension to use the above machinery. Over this field we will take a uniformly random affine transformation from $\mathbb{F}_q^n$ to $\mathbb{F}_q^m$. To create a field extension of $\mathbb{F}_{q_0}$, of size q_0^t, we use Rabin's algorithm (Lemma 2.13) to generate an irreducible polynomial of degree t, which gives an error probability of $\exp(-N/2t)$. To make this exponentially small in the input length we set $N = 2t \cdot (m \cdot s + 1)$.

For the affine transformation to be good with high probability, using Lemma 4.2 and [26], we need to pick $t \gg m \lg(\deg \overline{f}) + m \lg(m + \deg \overline{f}^{m-1})$. Recall that the maximum degree of any given f_i is 2^s. As the error probability needs to also be exponentially small in the output length we set $t = (m^2 s + m) + (ms + 1)$.

We run Rabin's algorithm with N and t described above. If it outputs an irreducible polynomial h, we construct $\mathbb{F}_q = \mathbb{F}_{q_0}[z]/\langle h \rangle$. Sample $A \in \mathbb{F}_q^{n \times m}$ and $\overline{\beta} \in \mathbb{F}_q^n$ uniformly, and output the algebraic circuit for $g_i(A\overline{y} + \overline{\beta})$ for all $i \in [m]$. If Rabin's algorithm fails to find an irreducible polynomial, we output a circuit computing the zero polynomial. In both cases we pad to ensure that the output is length-increasing.

Length-increasing. The reduction is always length increasing as we pad the output appropriately.

Complexity. Rabin's algorithm (Lemma 2.13) for finding irreducible polynomials runs in time $\mathsf{poly}(t, N, \lg q_0)$.

Sampling $A \in \mathbb{F}_q^{n \times m}$ and $\overline{\beta} \in \mathbb{F}_q^n$ uniformly and creating the n algebraic circuits described by $A\overline{y} + \overline{\beta}$ can be done in $\mathsf{poly}(m, t, \lg q_0)$. Constructing the circuits for $\overline{g}$ involves composing the circuits for $A\overline{y} + \overline{\beta}$ with each f_i and replacing all constants in the algebraic circuit for f_i by their image under the inclusion map $\mathbb{F}_{q_0} \hookrightarrow \mathbb{F}_q$. We can do this in $\mathsf{poly}(s, m, t, \lg q_0)$.

Lastly we can pad to make the output larger than the input in $\mathsf{poly}(s, m, \lg q_0)$ (Lemma 4.3).

All steps run in time $\mathsf{poly}(n, s, t, N, \lg q_0)$ and by the choice of t and N, this is polynomial in the input length.

Completeness. We need to show that if the $\overline{f}$ are algebraically independent, then the output is algebraically independent with high probability.

One source of error is that Rabin's algorithm fails to find an irreducible polynomial. The error probability of Rabin's algorithm is $\exp(-N/2t) \leq 2^{-ms-1}$.

Another source of error is that the polynomials $\overline{g}$, where $g_i = f_i(A\overline{y} + \overline{\beta})$, fail to be algebraically independent. From Lemma 4.2, we know that if $\overline{g}$ fail to be algebraically independent then there exists a polynomial $P \in \mathbb{F}_q[Z, \overline{w}]$ such that $P(A, \overline{\beta}) = 0$. Since $q_0 \geq 2$ and $\deg \overline{f} \leq 2^s$, by [26]

$$\Pr_{A,\overline{\beta}}\left[P(A, \overline{\beta}) = 0\right] \leq \frac{\deg P}{q} \leq \frac{(\deg(\overline{f}))^m (2\deg(\overline{f})^{m-1})^m}{q_0^t} \leq \frac{2^{m+m^2 s}}{2^t} \leq 2^{-ms-1}.$$

The total probability the reduction fails to output algebraically independent polynomials is at most $2^{-ms-1} + 2^{-ms-1} \leq 2^{-ms} \leq 2^{-\ell}$.

Soundness. We need to show that if the $\overline{f}$ are algebraically dependent then the output is always algebraically dependent.

If Rabin's algorithm fails, the output is the zero polynomial, which is algebraically dependent. If the reduction finds an irreducible polynomial, the output is $\overline{g}$ where $g_i = f_i(A\overline{y} + \overline{\beta})$. Let Q be any annihilator of $\overline{f}$, then $Q(\overline{g}) = Q(\overline{f}(A\overline{y} + \overline{\beta})) = 0$, thus $\overline{g}$ are algebraically dependent. ◀

By Lemma 2.16 and Corollary 3.11, since $\mathsf{AlgIndep}_= \in \mathsf{NISZK}$ and $\mathsf{AlgIndep}$ reduces to $\mathsf{AlgIndep}_=$ by a length-increasing randomized reduction with exponentially small error probability, we conclude that $\mathsf{AlgIndep} \in \mathsf{NISZK}$.

▶ **Corollary 4.5.** $\mathsf{AlgIndep} \in \mathsf{NISZK}$.

4.2 Changing the Number of Polynomials

Here we show that if we have m polynomials $f_1, \ldots, f_m$ of transcendence degree at least r, we can generate r linear combinations of $f_1, \ldots, f_m$ that will be algebraically independent with high probability. We show this by carefully following the proof of the Noether Normalization Lemma over infinite fields, found in [4]. Please refer to the full version for the proofs of the statements below.

▶ Remark 4.6. Our result, Corollary 4.11, is similar, but incomparable, to [9, Theorem 5.2.3]. Garg shows that with probability $1 - 2(r+1)(\deg \overline{f})^{m-1}/|S|$ the resulting r polynomials are algebraically independent where the coefficients of the linear combinations are sampled uniformly from $S \subset \mathbb{F} \setminus \{0\}$. We show, with probability $1 - (m-r)(\deg \overline{f})^{m-1}/|S|$, the r resulting polynomials are algebraically independent, and we give no restriction on the set S, however, the resulting linear combinations are created using S in a non-standard way.

Transcendence Degree

Throughout this section we will use an equivalent definition of transcendence degree in terms of fields, for more background see [16, Chapter 8], we omit the proof of its equivalence. Recall that a field extension $\mathbb{K}$ is *algebraic* over $\mathbb{F}$ if for every $\alpha \in \mathbb{K}$ there is a non-zero polynomial $P \in \mathbb{F}[z]$ with $P(\overline{\alpha}) = 0$.

▶ **Definition 4.7.** *Suppose $\mathbb{F} \subseteq \mathbb{K} \subseteq \mathbb{F}(\overline{x})$, the **transcendence degree** of $\mathbb{K}$ over $\mathbb{F}$ is the size of the smallest set $B \subset \mathbb{K}$ such that $\mathbb{K}$ is algebraic over $\mathbb{F}(B)$. The transcendence degree of $f_1, \ldots, f_m \in \mathbb{F}[\overline{x}]$, is the transcendence degree of $\mathbb{F}(\overline{f})$ over $\mathbb{F}$.*

This definition gives insight as to how transcendence degree behaves when we remove polynomials. If there exists an annihilator P for $\overline{f}$, such that $P(f_1, \ldots, f_{m-1}, y_m) \not\equiv 0$ in $\mathbb{F}(f_1, \ldots, f_{m-1})[y_m]$, then f_m is algebraic over $\mathbb{F}(f_1, \ldots, f_{m-1})$. If $\mathbb{F}(B) \subset \mathbb{F}(f_1, \ldots, f_{m-1})$ is algebraic, then the extension $\mathbb{F}(B) \subset \mathbb{F}(f_1, \ldots, f_m)$ is algebraic, and

$$\text{tr-deg}(f_1, \ldots, f_m) = \text{tr-deg}(f_1, \ldots, f_{m-1}).$$

When such a P exists, we say that f_m is *algebraically dependent* on $f_1, \ldots, f_{m-1}$. By definition, if $\mathbb{F}(\overline{f}) \subset \mathbb{F}(\overline{g})$ then $\text{tr-deg}(\overline{f}) \leq \text{tr-deg}(\overline{g})$.

In the proof of the Noether Normalization Lemma, found in [4], we begin with $m+1$ algebraically dependent polynomials $\overline{g}$, find a non-zero polynomial P where $P(\overline{\alpha}) \neq 0$ implies that taking the last polynomial g_{m+1} and subtracting $\alpha_i g_{m+1}$ from each $i \leq m$ preserves transcendence degree. We observe that the degree of the polynomial P is at most the degree of an annihilator, and thus the Perron bound (Lemma 2.4) gives an effective version of the Noether Normalization Lemma.

▶ **Proposition 4.8.** *If $g_1, \ldots, g_{m+1} \in \mathbb{F}[x_1, \ldots, x_n]$ are algebraically dependent with a non-zero annihilator of degree at most D, then there exists a non-zero $P \in \mathbb{F}[y_1, \ldots, y_m]$ with $\deg P \leq D$ such that if $P(\overline{\alpha}) \neq 0$ for $\overline{\alpha} \in \mathbb{F}^m$ then*

$$\text{tr-deg}(g_1 - \alpha_1 g_{m+1}, \ldots, g_m - \alpha_m g_{m+1}) = \text{tr-deg}(g_1, \ldots, g_{m+1}).$$

If we started with polynomials $f_1, \ldots, f_m$ algebraically dependent, applying Proposition 4.8 tells us something about the $m-1$ polynomials $\overline{h}$ given by the linear transformation $\overline{h} = A\overline{f}$ where $A = (I_{k-1} | -\overline{\alpha})$ is a $k-1 \times k$ matrix over $\mathbb{F}$. It will be useful to have notation for this linear transformation. For all k and $\overline{\gamma} \in \mathbb{F}^{k-1}$ let $A_k(\overline{\gamma})$ denote the matrix

$$A_k(\overline{\gamma}) := \Big(I_{k-1} \mid -\overline{\gamma}\Big).$$

If we repeatedly apply Proposition 4.8 $m-r$ times using vectors $\overline{\gamma}_m, \ldots, \overline{\gamma}_{r+1}$ where each $\overline{\gamma}_i \in \mathbb{F}^{i-1}$, the resulting polynomials are $(A_{r+1}(\overline{\gamma}_{r+1}) \cdots A_m(\overline{\gamma}_m))\overline{f}$.

We want algebraically independent polynomials from $\overline{f}$, so we only apply Proposition 4.8 $m - \text{tr-deg}(\overline{f})$ times on $\overline{f}$. Therefore, we can afford a union bound over each of these steps as the degree of an annihilator of $\overline{f}$ is significantly larger than $m - \text{tr-deg}(\overline{f})$ typically. This implies that if the vectors $\overline{\gamma}_i$ are chosen uniformly from a large enough set, with high probability the output is algebraically independent.

▶ **Lemma 4.9.** *If $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$ have transcendence degree r and $S \subseteq \mathbb{F}$, let $\Gamma = \{\overline{\gamma}\}_{i=r+1}^{m}$ and suppose each $\overline{\gamma}_k$ is chosen uniformly from S^{k-1}, then*

$$\Pr_{\Gamma}\left[\text{tr-deg}\left((A_{r+1}(\overline{\gamma}_{r+1})\cdots A_m(\overline{\gamma}_m))\overline{f}\right) < r\right] \leq \frac{(m-r)(\deg \overline{f})^r}{|S|}.$$

We can now verify that once we have algebraically independent polynomials, we can continue to left multiply by matrices $A_k(\overline{\gamma}_k)$ and the polynomials stay algebraically independent.

▶ **Proposition 4.10.** *Suppose $g_1, \ldots, g_m \in \mathbb{F}[x_1, \ldots, x_n]$ are algebraically independent and $r \leq m$, for any $\{\overline{\gamma}_i\}_{i=r+1}^{m}$ where $\overline{\gamma}_i \in \mathbb{F}^{i-1}$, the set $(A_{r+1}(\overline{\gamma}_{r+1})\cdots A_m(\overline{\gamma}_m))\overline{g}$ is algebraically independent.*

We conclude that if we started with polynomials of transcendence degree greater than r, with high probability the polynomials $(A_{r+1}(\overline{\gamma}_{r+1})\cdots A_m(\overline{\gamma}_m))\overline{f}$ are algebraically independent.

▶ **Corollary 4.11.** *Suppose $f_1, \ldots, f_m \in \mathbb{F}[\overline{x}]$, $\text{tr-deg}(\overline{f}) \geq r$, and $S \subseteq \mathbb{F}$, let $\Gamma = \{\overline{\gamma}_i\}_{i=r+1}^{m}$ and suppose $\overline{\gamma}_k$ are chosen uniformly from S^{k-1}, then*

$$\Pr_{\Gamma}\left[\text{tr-deg}\left((A_{r+1}(\overline{\gamma}_{r+1})\cdots A_m(\overline{\gamma}_m))\overline{f}\right) < r\right] \leq \frac{(m-r)(\deg \overline{f})^{m-1}}{|S|}.$$

The above corollary essentially gives the reduction from TrDeg to AlgIndep. We sketch the argument below.

On any input we will pass to a field extension, which we can construct via Rabin's algorithm (Lemma 2.13), to apply the above machinery. Over this large field we will take the random linear combinations described in Corollary 4.11 and output these polynomials. We note that these polynomials can be constructed efficiently in the input size. The degree of the field extension we pick is polynomial in the input size, therefore every step in the reduction runs efficiently.

If the input polynomials had high transcendence degree then we use Corollary 4.11 to argue that the probability the output is algebraically dependent is negligible. If the input polynomials had low transcendence degree then we output "too many" polynomials for them to be algebraically independent.

▶ **Theorem 4.12.** *There is a length-increasing randomized reduction with error probability $\delta \leq 2^{-\ell}$ on inputs of size ℓ from* TrDeg *to* AlgIndep*, that is,* $\text{TrDeg} \leq_m^{\text{BPP}} \text{AlgIndep}$.

By Lemma 2.16 and Corollary 4.5, $\text{AlgIndep} \in \text{NISZK}$ and TrDeg reduces to AlgIndep by a length-increasing randomized reduction with negligible error probability, we conclude $\text{TrDeg} \in \text{NISZK}$.

▶ **Corollary 4.13.** $\text{TrDeg} \in \text{NISZK}$.

References

1 Leonard M. Adleman and Hendrik W. Lenstra Jr. Finding irreducible polynomials over finite fields. In Juris Hartmanis, editor, *Proceedings of the 18th Annual ACM Symposium on Theory of Computing, May 28-30, 1986, Berkeley, California, USA*, pages 350–355. ACM, 1986. doi:10.1145/12130.12166.

2 Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. PRIMES is in P. *Annals of mathematics*, pages 781–793, 2004.

3 Eric Allender, Shuichi Hirahara, and Harsha Tirumala. Kolmogorov complexity characterizes statistical zero knowledge. In *Innovations in Theoretical Computer Science Conference*, 2023.

4 Allen Altman and Steven Kleiman. *A term of commutative algebra*. Worldwide Center of Mathematics, 2013.

5 László Babai and Shlomo Moran. Arthur-Merlin games: a randomized proof system, and a hierarchy of complexity classes. *Journal of Computer and System Sciences*, 36(2):254–276, 1988. doi:10.1016/0022-0000(88)90028-1.

6 Malte Beecken, Johannes Mittmann, and Nitin Saxena. Algebraic independence and blackbox identity testing. *Information and Computation*, 222:2–19, 2013. doi:10.1016/J.IC.2012.10.004.

7 Peter Bürgisser, Michael Clausen, and Mohammad A Shokrollahi. *Algebraic complexity theory*, volume 315. Springer Science & Business Media, 2013.

8 Zeev Dvir, Ariel Gabizon, and Avi Wigderson. Extractors and rank extractors for polynomial sources. *computational complexity*, 18(1):1–58, 2009. doi:10.1007/S00037-009-0258-4.

9 Abhibhav Garg. Special case algorithms for Nullstellensatz and transcendence degree. Master's thesis, IIT Kanpur, 2020.

10 Abhibhav Garg, Rafael Oliveira, and Akash Kumar Sengupta. Rank Bounds and PIT for $\Sigma^3\Pi\Sigma\Pi^d$ circuits via a non-linear Edelstein-Kelly theorem. *2025 IEEE 66th Annual Symposium on Foundations of Computer Science (FOCS)*, 2025.

11 Abhibhav Garg and Nitin Saxena. Special-case algorithms for blackbox radical membership, nullstellensatz and transcendence degree. In Ioannis Z. Emiris and Lihong Zhi, editors, *ISSAC '20: International Symposium on Symbolic and Algebraic Computation, Kalamata, Greece, July 20-23, 2020*, pages 186–193. ACM, 2020. doi:10.1145/3373207.3404030.

12 Oded Goldreich, Amit Sahai, and Salil Vadhan. Can statistical zero knowledge be made non-interactive? or on the relationship of szk and niszk. In *Annual International Cryptology Conference*, pages 467–484. Springer, 1999. doi:10.1007/3-540-48405-1_30.

13 Shafi Goldwasser and Michael Sipser. Private coins versus public coins in interactive proof systems. In *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, pages 59–68, 1986. doi:10.1145/12130.12137.

14 Zeyu Guo, Nitin Saxena, and Amit Sinhababu. Algebraic dependencies and pspace algorithms in approximative complexity over any field. *Theory of Computing*, 15(1):1–30, 2019. doi:10.4086/TOC.2019.V015A016.

15 Neeraj Kayal. The complexity of the annihilating polynomial. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 184–193. IEEE, 2009. doi:10.1109/CCC.2009.37.

16 Serge Lang. *Algebra*, volume 211. Springer Science & Business Media, 2012.

17 Johannes Mittmann, Nitin Saxena, and Peter Scheiblechner. Algebraic independence in positive characteristic: A p-adic calculus. *Transactions of the American Mathematical Society*, 366(7):3425–3450, 2014.

18 Tatsuaki Okamoto. On relationships between statistical zero-knowledge proofs. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 649–658, 1996. doi:10.1145/237814.238016.

19 Rafail Ostrovsky. One-way functions, hard on average problems, and statistical zero-knowledge proofs. In *Proceedings of the Sixth Annual Structure in Complexity Theory Conference, Chicago, Illinois, USA, June 30 - July 3, 1991*, pages 133–138. IEEE Computer Society, 1991. doi:10.1109/SCT.1991.160253.

20 James G Oxley. *Matroid theory*, volume 3. Oxford University Press, USA, 2006.

21 Anurag Pandey, Nitin Saxena, and Amit Sinhababu. Algebraic independence over positive characteristic: New criterion and applications to locally low-algebraic-rank circuits. *Comput. Complex.*, 27(4):617–670, 2018. doi:10.1007/S00037-018-0167-5.

22 Arkadiusz Płoski. Algebraic dependence of polynomials after o. perron and some applications. *Computational Commutative and Non-Commutative Algebraic Geometry*, pages 167–173, 2005.

23 Michael O Rabin. Probabilistic algorithms in finite fields. *SIAM Journal on computing*, 9(2):273–280, 1980. doi:10.1137/0209024.

24 Nitin Saxena and Comandur Seshadhri. Blackbox identity testing for bounded top fanin depth-3 circuits: the field doesn't matter. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 431–440, 2011.

25 Joachim Schmid. On the affine bezout inequality. *manuscripta mathematica*, 88(1):225–232, 1995.

26 Jacob T Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM (JACM)*, 27(4):701–717, 1980. doi:10.1145/322217.322225.

27 Victor Shoup. *A computational introduction to number theory and algebra*. Cambridge university press, 2009.

28 Amir Shpilka, Amir Yehudayoff, et al. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends® in Theoretical Computer Science*, 5(3–4):207–388, 2010. doi:10.1561/0400000039.