# Randomized Satisfiability Checking for Non-Linear Arithmetic over Finite Fields

Amir Goharshady
University of Oxford
Oxford, UK
amir.goharshady@cs.ox.ac.uk

Thomas Hader
TU Wien
Vienna, Austria
thomas.hader@tuwien.ac.at

Laura Kovács
TU Wien
Vienna, Austra
laura.kovacs@tuwien.ac.at

Harshit Jitendra Motwani
TU Wien
Vienna, Austria
harshit.motwani@tuwien.ac.at


***Abstract*—We address the satisfiability problem in the theory of non-linear arithmetic over finite fields. This problem is of significant practical interest, particularly motivated by verification tasks from cryptography, coding theory, and smart contracts. While existing methods provide completeness, they can become impractical when the solution set has non-zero dimension and brute-force enumeration is infeasible for the field sizes that arise in practice. We propose a randomized algorithm that repeatedly intersects the polynomial system with uniformly random affine linear constraints (hyperplane slices) to progressively reduce the effective dimension of the solution space. Once a zero-dimensional system is obtained, we invoke existing techniques from finite field SMT solving to obtain a model. If the input is satisfiable, we find a witness with high probability, with success amplified by independent trials. We formalize the dimension-drop intuition via probabilistic slicing arguments over finite fields and show how our method fits naturally into an SMT workflow. We implemented our algorithm as an improvement to the finite field reasoning engine of the CVC5 SMT solver. We provide extensive experimental results demonstrating significant gains in scalability when using our work compared to existing methods implemented in CVC5.**

## I. Introduction

Non-linear arithmetic is ubiquitous in automated reasoning and formal verification, especially when considering applications arising from physical dynamics, control theory, robotics, smart contracts, finance, and other models [35], [25], [3], [33]. In particular, to assess the functional correctness of programs with non-linear updates or guards, we need to reason about non-linear arithmetic. Several safety-verification tasks reduce to checking satisfiability of non-linear arithmetic formulas. The underlying theory and, in particular, the choice of algebraic domains, such as fields, depend on the application domain. In this paper we focus on non-linear arithmetic over finite fields, which are commonly used to model problems in cryptography and blockchain.

***Non-linear arithmetic and SMT.*** For physical systems, control, and robotics, we typically reason about real arithmetic. SMT solving for non-linear real arithmetic has therefore received extensive attention over the past decades. Modern solvers such as CVC5 [4], Z3 [8], and YICES2 [9] can handle many such formulas efficiently in practice, using techniques such as cylindrical algebraic decomposition (CAD) [6], Gröbner basis computations, positivity theorems, and sum-of-squares reasoning [34]. Overall, satisfiability over the reals is well-studied both theoretically and practically, and there is a rich set of benchmarks from real-world applications.

On the other hand, programs arising in cryptography [22], coding theory [29], smart contracts [39], and Zero-Knowledge Proofs [17], [33] often require reasoning over finite fields, also called Galois fields [14] and typically written as $\mathbb{F}_q$, whose size $q$ is a prime power; that is, $q = p^t$, where $p$ is a prime number and $t > 0$ is a positive integer. Verification tasks in these domains again reduce to satisfiability and quantifier elimination for non-linear formulas over finite fields, most commonly over $\mathbb{F}_{2^n}$.

However, in contrast to non-linear real arithmetic, satisfiability over finite fields is much less explored, both theoretically and practically. Only a few SMT solvers currently support finite field reasoning, such as CVC5 [31], [32] and YICES2 [20], and this support has only been added relatively recently. One fundamental reason for this gap is that, unlike real arithmetic, finite fields lack a natural ordering, and hence notions such as positivity, limits, or continuity. This rules out many of the workhorse techniques for non-linear real arithmetic, including CAD (which fundamentally relies on ordering and sign-invariance), interval-based reasoning, and sum-of-squares or positivity arguments. Consequently, reasoning over finite fields is necessarily more algebraic in nature. Moreover, even when employing algebraic techniques, such as Gröbner basis methods [5], the finite field setting differs fundamentally from the familiar cases of rationals $\mathbb{Q}$, reals $\mathbb{R}$, or complex numbers $\mathbb{C}$. Finite fields have positive characteristic and are finite domains, which introduces additional algebraic identities and typically requires incorporating field polynomials such as $x^q - x$ to restrict solutions to $\mathbb{F}_q$. As a result, the behavior

https://doi.org/10.34727/2026/isbn.978-3-85448-093-8_14

of polynomial systems over finite fields can differ markedly from the real or complex setting.

We also emphasize that working over finite fields does not necessarily make the problem easier, as finiteness can still lead to very large field sizes. Satisfiability over finite fields is decidable and admits complete decision procedures, unlike the often incomplete or heuristic methods used over the reals. However, brute-force enumeration is infeasible for the field sizes that arise in practice, with up to 256-bit primes used in cryptographic applications. This creates a need for algorithms that can reason efficiently about non-linear arithmetic over finite fields while avoiding exhaustive search over all assignments.

Existing approaches in the literature combine techniques ranging from bit-blasting [33] to Gröbner basis computations over field polynomials [15]. In practice, these methods either do not scale well or sacrifice completeness. Moreover, guess-and-check strategies are used when the solution space has positive dimension [31]; however, due to the large sizes of fields, such strategies can still be unscalable across diverse problem instances.

***Our contributions.*** We propose a *randomized approach* that adds random linear constraints (hyperplanes) to intersect the corresponding ideal and progressively reduce the effective dimension of the solution space (see Section IV). Once the reduced dimension $d$ reaches zero, we solve the resulting zero-dimensional system by computing a Gröbner basis, where the computation is substantially more efficient (see Algorithm 1). Each dimensionality-reduction step succeeds with a certain probability (see Section IV-A). Thus, if the input formula is satisfiable, our method finds a solution with high probability. We implemented our algorithm on top of the CVC5 solver and demonstrate its practical scalability on a number of examples (see Section V). Additionally, we introduce a Jacobian-based heuristic for estimating the dimension of polynomial ideals over finite fields, which in practice produces accurate estimates in polynomial time (see Sections IV-B and V).

## II. Illustrative Example

We illustrate the main idea of our approach on the following example, using a finite field $\mathbb{F}_q$ with a large prime power $q$. We denote by $\mathbb{F}_q^*$ the multiplicative group of nonzero elements in $\mathbb{F}_q$.

***A positive-dimensional system with many solutions.*** Consider the non-linear constraint

$$xy = 1 \qquad \text{over } \mathbb{F}_q. \tag{1}$$

The polynomial system (1) has algebraic dimension 1, as the maximal number of variables that are independent modulo the ideal defined by $xy = 1$ is 1. Intuitively, for a polynomial system of effective dimension $d$ over $\mathbb{F}_q$, one expects $\mathcal{O}(q^d)$ solutions (by Lang-Weil estimates [28], [40]). This intuition is exact in the present case: the polynomial (1) has exactly $q - 1$ solutions in $\mathbb{F}_q^2$, since every choice of $x \in \mathbb{F}_q^*$ uniquely determines $y = x^{-1}$. For cryptographic field sizes $q$ modulo 256-bit primes, such a solution set is far too large to enumerate. We overcome this burden by computing solutions from slicing with hyperplanes induced by linear constraints.

***Slicing with a hyperplane.*** Let us add a linear constraint

$$ax + by + c = 0, \tag{2}$$

where $a, b, c \in \mathbb{F}_q$ and $(a, b) \neq (0, 0)$. Intersecting (1) with (2) yields

$$xy = 1 \qquad \wedge \qquad ax + by + c = 0. \tag{3}$$

Substituting $y = x^{-1}$ into (2) gives

$$ax + bx^{-1} + c = 0.$$

Since $x \neq 0$ for any solution of $xy = 1$, we can multiply by $x$ and obtain the univariate quadratic equation

$$ax^2 + cx + b = 0. \tag{4}$$

When $a \neq 0$, this equation has at most two solutions in $\mathbb{F}_q$, and each such solution uniquely determines a value of $y$. When $a = 0$ and $b \neq 0$, the linear constraint fixes $x$ to a single value and the system has at most one solution. In either of the cases, adding the hyperplane (2) reduces the number of solutions of (1) from $q - 1$, in our case of order $q^1$, to a constant, that is to $q^0$. Equivalently, the resulting (sliced) system (3) is zero-dimensional.

***Genericity and randomization.*** The above example also illustrates what we mean by adding a generic linear constraint, that is a hyperplane. Certain choices of the coefficients $a, b, c$ may eliminate all solutions even when (1) is satisfiable; for example, when the hyperplane fails to intersect the solution set over $\mathbb{F}_q$. Our approach therefore samples the coefficients of the linear constraint randomly, subject to $(a, b) \neq (0, 0)$, and repeats the slicing step as needed. Intuitively, for a positive-dimensional system, a randomly chosen hyperplane reduces the effective dimension with non-negligible probability, dramatically shrinking the solution space. Once the dimension reaches zero, the resulting system can be solved efficiently using Gröbner basis computation, as the number of solutions is constant and also the time-complexity of computing a Gröbner basis is single exponential in the number of variables for zero-dimensional systems. We formalize this intuition via probabilistic slicing arguments over finite fields in Section IV.

## III. Preliminaries

We recall standard notions from algebraic geometry over finite fields and refer to [7], [23] for details.

Throughout, $k$ denotes a field and $\overline{k}$ its algebraic closure. We write $\mathbb{F}_q$ for the finite field of size $q = p^t$ with $p$ prime and integer $t > 0$. For $n \in \mathbb{N}$, let $k[X] = k[x_1, \ldots, x_n]$ be the polynomial ring over the variables $x_1, \cdots, x_n$. For $f_1, \ldots, f_m \in k[X]$, we write $\langle f_1, \ldots, f_m \rangle$ for the ideal they generate.

***Affine varieties and Zariski topology.*** For an ideal $I \subseteq k[X]$, its affine variety is defined as:

$$V(I) \;:=\; \{a \in k^n : f(a) = 0 \text{ for all } f \in I\}.$$

A subset $Z \subseteq k^n$ is *(Zariski) closed* if $Z = V(I)$ for some ideal $I$. These closed sets define the *Zariski topology* on $k^n$. For $S \subseteq k^n$, the *vanishing ideal* of $S$ is

$$I(S) \;:=\; \{f \in k[X] : f(a) = 0 \text{ for all } a \in S\},$$

and the *Zariski closure* of $S$ is $\overline{S} := V(I(S))$.

***Coordinate rings and radicals.*** The *coordinate ring* of an affine variety $V \subseteq k^n$ is $k[V] := k[X]/I(V)$. An ideal $I$ is *radical* if $I = \sqrt{I} := \{f : f^t \in I \text{ for some } t > 0\}$. Over algebraically closed fields, Hilbert's Nullstellensatz implies $I(V(I)) = \sqrt{I}$. Over $\mathbb{F}_q$, it is convenient to consider a base change to $\overline{\mathbb{F}}_q$.

***Finite fields.*** Let $\mathbb{F}_q$ be the finite field with $q = p^t$ elements. Every $a \in \mathbb{F}_q$ satisfies $a^q = a$, and $\mathbb{F}_q^\times$ is multiplicative group of non-zero elements in $\mathbb{F}_q$. The algebraic closure is $\overline{\mathbb{F}}_q = \bigcup_{m \geq 1} \mathbb{F}_{q^m}$, and $\mathbb{F}_q$ is exactly the set of roots of $x^q - x$ in $\overline{\mathbb{F}}_q$. For a variety (solution set) $V$ defined over $\mathbb{F}_q$, we write $V(\mathbb{F}_q)$ for its $\mathbb{F}_q$-rational points.

***Irreducibility and components.*** A variety $V$ is *irreducible* if it cannot be written as the union of two proper closed subsets. Equivalently, $V$ is irreducible iff $I(V)$ is prime (over an algebraically closed base). Every variety $V$ admits a decomposition into finitely many *irreducible components*

$$V = V_1 \cup \cdots \cup V_r,$$

where each $V_i$ is irreducible and $V_i \nsubseteq V_j$ for $i \neq j$.

***Geometric irreducibility.*** A variety $V$ defined over $\mathbb{F}_q$ is *geometrically irreducible* if $V_{\overline{\mathbb{F}}_q} := V \times_{\mathbb{F}_q} \overline{\mathbb{F}}_q$ is irreducible as a variety over $\overline{\mathbb{F}}_q$. In other words, $V$ is geometrically irreducible iff it remains irreducible over the base change to $\overline{\mathbb{F}}_q$.

***Dimension.*** For an affine variety $V$ (with its Zariski topology), the *dimension* is the largest integer $d$ for which there exists a strict chain of irreducible closed subsets

$$Z_0 \subsetneq Z_1 \subsetneq \cdots \subsetneq Z_d \subseteq V.$$

Equivalently,

$$\dim(V) \;:=\; \sup\big\{\, d \in \mathbb{N} \;\big|\; \exists\, Z_0 \subsetneq \cdots \subsetneq Z_d \subseteq V \text{ irreducible and closed} \,\big\}.$$

For reducible $V = \bigcup_i V_i$, one has $\dim(V) = \max_i \dim(V_i)$. For affine varieties, this notion agrees with the Krull dimension of the coordinate ring [2].

***Affine hyperplanes.*** An affine hyperplane in $k^n$ is a set

$$H_{a,b} \;:=\; \{x \in k^n : \langle a, x\rangle = b\},$$

where $a \in k^n \setminus \{0\}$ and $b \in k$, and $\langle a, x\rangle$ denotes the standard inner product. Equivalently, $H_{a,b} = V(\langle \langle a, x\rangle - b\rangle)$.

Geometrically, an affine hyperplane can be viewed as the set of solutions to a linear equation of the form $\sum_i a_i x_i - b = 0$, where $a_i$ and $b$ are elements of $k$. Intersecting a variety $V$ with a hyperplane $H_{a,b}$ corresponds to adding the linear polynomial $\sum_i a_i x_i - b$ to the set of polynomials defining the ideal of $V$. This operation restricts the solution set to those points in $V$ that also satisfy the given linear constraint.

## IV. Randomized Algorithm for Satisfiability over Finite Fields

We present our Algorithm 1 for *solving polynomial satisfiability over finite fields* using random hyperplane slicing. The algorithm iteratively adds random affine linear constraints to the input system (line 4 of Algorithm 1), effectively slicing the solution space and reducing its dimension (line 5). Once a zero-dimensional system is obtained, we invoke an oracle to find a witness (line 6). If the input is satisfiable, our algorithm finds a witness with high probability (line 10), which can be amplified by independent trials. By reducing the dimension of the input system, the oracle may assume a zero-dimensional system with high probability.

We present the mathematical foundations behind our Algorithm 1 in Section IV-A, establishing that intersecting with a random hyperplane reduces the dimension of the solution space with high probability. Additionally, Section IV-B introduces a Jacobian-based heuristic for efficiently estimating the dimension of the variety defined by the input polynomials, which is essential for determining how many random hyperplanes to add in practice. Finally, Section IV-C details the oracle $\mathcal{O}_{\mathbb{F}_q}$ used to solve the resulting zero-dimensional systems.

**Algorithm 1** Randomized hyperplane slicing for polynomial satisfiability over $\mathbb{F}_q$

**Require:** Polynomials $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$; (upper bound on) dimension $d$; number of trials $T$
**Ensure:** Either SAT with a witness $x \in \mathbb{F}_q^n$, or UNKNOWN
1: **for** $t = 1$ to $T$ **do**
2: Sample $A \in \mathbb{F}_q^{d \times n}$ uniformly at random conditioned on $\operatorname{rank}(A) = d$
3: Sample $b \in \mathbb{F}_q^d$ uniformly at random
4: Define affine linear forms $\ell_i(x) := \sum_{j=1}^n A_{ij} x_j - b_i$ for $i = 1, \ldots, d$
5: Form the sliced ideal
$$J_t \;:=\; \langle f_1, \ldots, f_m, \ell_1, \ldots, \ell_d\rangle \;\subseteq\; \mathbb{F}_q[x_1, \ldots, x_n].$$
6: Use oracle $\mathcal{O}_{\mathbb{F}_q}$ to find $x^{(t)} \in \mathbb{F}_q^n$ satisfying
$$f_1(x) = \cdots = f_m(x) = \ell_1(x) = \cdots = \ell_d(x) = 0,$$
or return NO-SOLUTION
7: **if** $\mathcal{O}_{\mathbb{F}_q}$ returns NO-SOLUTION **then**
8: **continue**
9: **else if** $f_k(x^{(t)}) = 0$ for all $k \in \{1, \ldots, m\}$ **then**
10: **return** (SAT, $x^{(t)}$)
11: **end if**
12: **end for**
13: **return** UNKNOWN

### A. Dimensionality Reduction via Random Hyperplane Slicing

We now present the mathematical foundation behind Algorithm 1. In particular, we show that Algorithm 1 drops

the dimension of a variety by one with high probability by intersecting it with a random hyperplane.

The following results are inspired by the classical Bertini theorems [24], [36] in algebraic geometry. Here, we establish a probabilistic version tailored to finite fields, which is useful for the dimensionality reduction step in our randomized algorithm.

**Theorem 1** (Probabilistic Bertini Theorem over Finite Fields)**.** *Let $V \subseteq \mathbb{F}_q^n$ be a geometrically irreducible affine variety of dimension $d \geq 2$, defined over a finite field $\mathbb{F}_q$. Then there exists a constant $K > 0$, depending only on the degree and defining complexity of $V$, such that the probability that a uniformly random affine hyperplane $H \subseteq \mathbb{F}_q^n$ intersects $V$ in an irreducible subvariety of dimension exactly $d-1$ is at least*

$$1 - \frac{K}{q}.$$

*In particular, the set of affine hyperplanes $H$, such that $\dim(V \cap H) = d-1$ and $V \cap H$ remains irreducible, forms a $1 - \mathcal{O}(1/q)$ fraction of all affine hyperplanes over $\mathbb{F}_q$.*

*Proof.* Let $\mathcal{H}$ denote the parameter space of affine hyperplanes in $\mathbb{F}_q^n$. We identify $\mathcal{H}$ with the set of pairs $(a, b) \in \mathbb{F}_q^n \setminus \{0\} \times \mathbb{F}_q$, where the corresponding hyperplane is $H_{a,b} = \{x \in \mathbb{F}_q^n : \langle a, x\rangle = b\}$.

Let $\overline{\mathbb{F}}_q$ be the algebraic closure of $\mathbb{F}_q$, and consider the base change $V_{\overline{\mathbb{F}}_q} \subseteq \overline{\mathbb{F}}_q^n$. Since $V$ is geometrically irreducible over $\mathbb{F}_q$, it remains irreducible over $\overline{\mathbb{F}}_q$, and we may apply classical Bertini's theorem over $\overline{\mathbb{F}}_q$.

By Bertini's theorem, the set of affine hyperplanes $H$ over $\overline{\mathbb{F}}_q$ such that $V_{\overline{\mathbb{F}}_q} \cap H$ fails to be irreducible or fails to drop dimension is contained in a proper Zariski-closed subset $\mathcal{B} \subsetneq \mathcal{H}_{\overline{\mathbb{F}}_q}$. This means that for generic $H$ over $\overline{\mathbb{F}}_q$, the intersection $V \cap H$ is irreducible of dimension $d-1$.

Now, we consider the $\mathbb{F}_q$-rational points of $\mathcal{H}$, i.e., the affine hyperplanes defined over the base field. Suppose for contradiction that all such hyperplanes lie in the bad set $\mathcal{B}$. Then $\mathcal{H}(\mathbb{F}_q) \subseteq \mathcal{B}$. But $\mathcal{H}$ is irreducible of dimension $n$, while $\mathcal{B}$ is a proper closed subvariety of dimension strictly less than $n$, so by standard point-counting estimates (e.g., Lang–Weil type bounds [28], [40]), we have

$$|\mathcal{B}(\mathbb{F}_q)| = \mathcal{O}(q^{n-1}), \quad |\mathcal{H}(\mathbb{F}_q)| = \mathcal{O}(q^n),$$

which contradicts the assumption that all rational hyperplanes are bad. Therefore, there exists at least one hyperplane over $\mathbb{F}_q$ outside of $\mathcal{B}$.

Moreover, since $\mathcal{B}$ is a Zariski-closed proper subset, the density of bad hyperplanes among $\mathbb{F}_q$-rational ones is at most $\mathcal{O}(1/q)$, so the desired property (dimension drops and irreducibility is preserved) holds with probability at least $1 - \mathcal{O}(1/q)$. □

**Remark 1.** *Note that the constant $K$ in Theorem 1 depends on the complexity of the variety $V$, which includes the number of variables and the degree of the variety, irrespective of the size of the finite field $\mathbb{F}_q$. Since we are intersecting $V$ with generic hyperplanes, the new variety $V \cap H$ will not have higher complexity than $V$ itself. Therefore, we can repeat the process of intersecting the resulting varieties with hyperplanes until we reach a variety of dimension 1 with high probability. We further note that Theorem 1 assumes that the variety $V$ is geometrically irreducible, which means that $V$ remains irreducible over the algebraic closure $\overline{\mathbb{F}}_q$ of the finite field $\mathbb{F}_q$. Moreover, in applications $q$ is large and the probability increases for $q \gg K$.*

As the Bertini-type theorems are typically stated for varieties of dimension at least 2, we need a separate argument to handle the case when the variety has dimension 1. The following theorem addresses this case, showing that intersecting a curve with a random hyperplane reduces its dimension to zero with high probability, although the resulting variety may be reducible. Theorem 2 thus provides a sufficient condition for our purposes, as we only need to find a single solution for satisfiability checking.

**Theorem 2** (Dimension Drop for Affine Curves over Finite Fields)**.** *Let $C \subseteq \mathbb{F}_q^n$ be an affine algebraic curve, that is, a (Zariski) closed subset with $\dim C = 1$, such that $C_{\overline{\mathbb{F}}_q}$ is irreducible. Let $A \subseteq \overline{\mathbb{F}}_q^n$ be the smallest affine linear subspace containing $C_{\overline{\mathbb{F}}_q}$, and define $r := \dim A$. Let $H \subseteq \mathbb{F}_q^n$ be a uniformly random affine hyperplane. Then*

$$\Pr[\dim(C \cap H) = 0] \;\geq\; 1 - \frac{q^{n-r}-1}{q\,(q^n-1)} \;\geq\; 1 - \mathcal{O}\Big(q^{-(r+1)}\Big).$$

*where* Pr *denotes the probability taken over the choice of affine hyperplane $H$.*

*Proof.* Any affine hyperplane $H \subseteq \mathbb{F}_q^n$ can be written as $H = H_{a,b} := \{x \in \mathbb{F}_q^n : \langle a, x\rangle = b\}$ for some $a \in \mathbb{F}_q^n \setminus \{0\}$ and $b \in \mathbb{F}_q$. We will also view the same equation over $\overline{\mathbb{F}}_q$, namely

$$H(\overline{\mathbb{F}}_q) := \{x \in \overline{\mathbb{F}}_q^n : \langle a, x\rangle = b\},$$

so that set containments involving $C_{\overline{\mathbb{F}}_q}$ and $A$ are interpreted inside $\overline{\mathbb{F}}_q^n$.

Since $C_{\overline{\mathbb{F}}_q}$ is irreducible of dimension 1, the intersection $C \cap H$ is a closed subset of $C$ and hence has dimension at most 1. Moreover,

$$\dim(C \cap H) = 1 \quad \Longleftrightarrow \quad C_{\overline{\mathbb{F}}_q} \subseteq H(\overline{\mathbb{F}}_q).$$

Indeed, if $\dim(C \cap H) = 1$, then $C \cap H$ is a closed subset of the irreducible curve $C$ with the same dimension, and therefore equals $C$.

By definition, $A$ is the smallest affine linear subspace of $\overline{\mathbb{F}}_q^n$ containing $C_{\overline{\mathbb{F}}_q}$. Since $H(\overline{\mathbb{F}}_q)$ is itself an affine linear subspace, we have

$$C_{\overline{\mathbb{F}}_q} \subseteq H(\overline{\mathbb{F}}_q) \quad \Longleftrightarrow \quad A \subseteq H(\overline{\mathbb{F}}_q).$$

Thus, the "bad" event (no dimension drop) implies that $H$ contains $A$ after extending scalars to $\overline{\mathbb{F}}_q$.

Write $A = x_0 + W$ with $x_0 \in \overline{\mathbb{F}}_q^n$ and $W \subseteq \overline{\mathbb{F}}_q^n$ an $r$-dimensional linear subspace. If $A \subseteq H(\overline{\mathbb{F}}_q)$ for $H = H_{a,b}$, then for all $w \in W$ we must have $\langle a, w\rangle = 0$, i.e., the linear

form $x \mapsto \langle a, x\rangle$ annihilates $W$. The set of such $a$ forms an $\mathbb{F}_q$-linear subspace of $\mathbb{F}_q^n$ of dimension at most $n-r$, hence there are at most $q^{n-r}-1$ nonzero choices of $a$ and therefore at most

$$\frac{q^{n-r}-1}{q-1}$$

possible normal *directions* up to scaling.

For each such direction $[a]$, there is *at most one* $b \in \mathbb{F}_q$ for which $H_{a,b}(\overline{\mathbb{F}}_q)$ contains $A$ (namely $b = \langle a, x_0\rangle$, if this value happens to lie in $\mathbb{F}_q$). In particular, this yields the upper bound

$$\#\{H \text{ affine hyperplane over } \mathbb{F}_q : C \subseteq H\} \;\le\; \frac{q^{n-r}-1}{q-1}.$$

We emphasize that if $A$ is not defined over $\mathbb{F}_q$, then for many directions $[a]$ one may have $a \cdot x_0 \notin \mathbb{F}_q$, in which case no $\mathbb{F}_q$-hyperplane with that normal direction contains $A$; thus the bound above may be strict.

The total number of affine hyperplanes in $\mathbb{F}_q^n$ equals

$$q \cdot \frac{q^n-1}{q-1}.$$

Therefore,

$$\Pr[C \subseteq H] \;\le\; \frac{\frac{q^{n-r}-1}{q-1}}{q \cdot \frac{q^n-1}{q-1}} = \frac{q^{n-r}-1}{q(q^n-1)}.$$

Since $\dim(C \cap H) = 1$ if and only if $C \subseteq H$, we conclude

$$\Pr[\dim(C \cap H) = 0] = 1 - \Pr[C \subseteq H] \;\ge\; 1 - \frac{q^{n-r}-1}{q(q^n-1)}.$$

The asymptotic bound follows immediately. □

**Remark 2.** *The parameter $r$ is defined geometrically as the dimension of the smallest affine linear subspace containing $C_{\overline{\mathbb{F}}_q}$. Since $C_{\overline{\mathbb{F}}_q}$ has dimension 1, it cannot be contained in a single point, and therefore $r \ge 1$ always holds.*

Combining the procedure outlined above with Theorem 2, we get a high probability to reach a zero-dimensional ideal $J'$ by adding one more linear form $\ell_d$. This results in the the procedure outlined in Algorithm 1 which allows the oracle $\mathcal{O}_{\mathbb{F}_q}$ to assume zero-dimension, enabling it to use more efficient procedures for checking for a common zero in the ideal, i.e. checking satisfiability of the polynomial set.

***Reducible varieties.*** When $V$ is reducible, we write its decomposition over $\overline{\mathbb{F}}_q$ as $V_{\overline{\mathbb{F}}_q} = V_1 \cup \cdots \cup V_t$ into irreducible components. In this case, a random hyperplane slice can fail to reduce the dimension if it contains at least one component $V_i$ of maximal dimension; this is so because $V_i \subseteq V \cap H$ and $\dim(V \cap H) = \dim(V)$. For each fixed component $V_i$, the set of hyperplanes $H$ with $V_i \subseteq H$ forms a proper (Zariski) closed subset of the hyperplane parameter space; hence, by the same point-counting/Lang–Weil argument [28], [40] as in the irreducible case, a uniformly random affine hyperplane contains $V_i$ with probability $O(1/q)$, with constants depending only on the complexity of $V_i$. Applying a union bound over the $t$ components, the probability that a random hyperplane contains some $V_i$ and thus does not reduce dimension is at most $O(t/q)$. Consequently, even for reducible $V$, intersecting with a random hyperplane reduces the dimension with high probability and the dependence on reducibility can be absorbed into the overall complexity parameter.

***Probability of success.*** As established by the above theorems, a single execution of the randomized slicing procedure succeeds with probability at least $1 - (d \cdot C)/q$, where $d$ is the dimension of the input variety and $C$ is a constant depending on its geometric complexity (e.g., degree, number of variables, and number of irreducible components). In the worst case, this complexity parameter can be large and may grow exponentially with the input size. Accordingly, we do not claim any improvement over the worst-case complexity of standard Gröbner basis algorithms. Rather, our contribution is a randomized approach whose probability of success increases with the field size $q$ and which exhibits significantly improved average-case and practical performance on instances of interest.

***Estimating dimensions.*** To apply our randomized slicing approach, we require an estimate of the dimension of the variety. In some cases, the dimension may be known by construction or inferred from auxiliary properties of the problem. Generally, however, an efficient heuristic for dimension estimation is desirable; the next section describes a Jacobian-based heuristic for this purpose. If such a heuristic fails or is unavailable, one can resort to a trial-and-error strategy, such as a binary search over the number of hyperplanes added. Adding too many hyperplanes may eliminate all solutions, while adding too few typically results in a positive-dimensional system, though the oracle may still find a solution if one exists. In practice, we can start with a small number of hyperplanes and incrementally increase it until a solution is found or a predetermined threshold is reached.

### *B. Jacobian-based Heuristic for Estimating the Dimension*

Computing the exact dimension of an algebraic variety defined by polynomial constraints is a computationally difficult problem in general. Classical results show that dimension computation via Gröbner bases can require doubly exponential time in the worst case. Over finite fields, related problems such as counting the number of solutions of a polynomial system are #P-hard [41]. Moreover, by the Lang–Weil estimates [28], [40], the dimension of a variety governs the asymptotic growth of the number of $\mathbb{F}_q$-rational points, with $|V(\mathbb{F}_q)| = q^d + O(q^{d-1/2})$ for irreducible varieties. These results suggest that exact dimension computation is unlikely to admit a scalable deterministic algorithm that works uniformly for all inputs.

In light of this inherent complexity, we adopt a *lightweight heuristic approach that estimates the dimension under genericity assumptions* and is sufficient for guiding our randomized slicing strategy, while preserving soundness of satisfiability results.

Let $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ be the input polynomials, and let

$$J(x) = \left( \frac{\partial f_i}{\partial x_j}(x) \right)_{1 \le i \le m,\ 1 \le j \le n}$$

denote their Jacobian matrix. The entries of $J$ are polynomials over $\mathbb{F}_q$, so $J(a)$ can be evaluated at any $a \in \mathbb{F}_q^n$, regardless of whether $a$ is a solution.

Our procedure samples a small number of points $a^{(1)}, \ldots, a^{(s)} \in \mathbb{F}_q^n$ uniformly at random and computes the rank of each $J(a^{(t)})$. We then set

$$\hat{r} := \max_{t=1}^{s} \operatorname{rank}\left(J(a^{(t)})\right),$$

and estimate the dimension as $\hat{d} := n - \hat{r}$.

***Probabilistic reasoning.*** Let $r^* := \max_{a \in \mathbb{F}_q^n} \operatorname{rank}(J(a))$ denote the *generic Jacobian rank*, i.e., the largest integer such that some $r^* \times r^*$ minor of $J(x)$ is a nonzero polynomial. Let $D$ be an upper bound on the total degree of the input polynomials, so each entry of $J$ has degree at most $D-1$, and any $r^* \times r^*$ minor has degree at most $r^*(D-1)$.

For a uniformly random $a \in \mathbb{F}_q^n$, the rank of $J(a)$ is less than $r^*$ only if all nonzero $r^* \times r^*$ minors vanish at $a$. By the Schwartz–Zippel lemma [37], [43],

$$\Pr[\operatorname{rank}(J(a)) < r^*] \le \frac{r^*(D-1)}{q}.$$

Thus, after $s$ independent samples,

$$\Pr[\hat{r} < r^*] \le \left( \frac{r^*(D-1)}{q} \right)^s,$$

so with high probability,

$$\Pr[\hat{r} = r^*] \ge 1 - \left( \frac{r^*(D-1)}{q} \right)^s.$$

The above description is inherently heuristic. Its effectiveness relies on standard genericity assumptions on the input system, such as equidimensionality, absence of hidden algebraic dependencies among constraints, and generic smoothness of the resulting variety. Under these conditions, the defining equations behave like independent constraints, and the generic rank of the Jacobian coincides with the geometric codimension of the variety. Consequently, we intuitively expect the estimated ambient dimension to match the true dimension in such cases. Importantly, these assumptions are not required for soundness of the overall procedure: when they do not hold, the dimension estimate may be inaccurate, in this scenario, we may end up adding more or fewer hyperplanes than necessary, but this does not affect the correctness of the algorithm, as we can always verify any candidate solution returned by the oracle.

### C. Oracle for Zero-Dimensional Polynomial Systems over Finite Fields

We next discuss the oracle $\mathcal{O}_{\mathbb{F}_q}$ used in Algorithm 1 for solving polynomial systems over finite fields. The role of $\mathcal{O}_{\mathbb{F}_q}$ is as follows. Given the sliced system as in line 6 of Algorithm 1

$$f_1(x) = \cdots = f_m(x) = \ell_1(x) = \cdots = \ell_d(x) = 0 \quad \text{over } \mathbb{F}_q,$$

the oracle $\mathcal{O}_{\mathbb{F}_q}$ should either return a witness $x \in \mathbb{F}_q^n$ or report NO-SOLUTION. Algorithm 1 only needs one-sided behavior: if a solution is returned, we can verify it directly. The oracle $\mathcal{O}_{\mathbb{F}_q}$ may be incomplete.

Based on the arguments above, after intersecting with random hyperplanes we can hope to obtain a *zero-dimensional* variety with high probability. Once we are in the zero-dimensional setting, there are known algorithms for solving polynomial systems over finite fields efficiently. In particular, while general Gröbner basis computation can be doubly exponential in the number of variables in the worst case, the zero-dimensional case admits algorithms with single-exponential behavior in the number of variables [26], [27]. This is exactly the complexity advantage we want to exploit for the sliced system.

Towards constructing an oracle $\mathcal{O}_{\mathbb{F}_q}$ we start by computing a Gröbner basis $G_t$ for the sliced ideal $J_t = \langle f_1, \ldots, f_m, \ell_1, \ldots, \ell_d \rangle$ using an algorithm tuned for the zero-dimensional setting. In case we can deduce $1 \in J_t$ we return NO-SOLUTION right away. Using $G_t$ we can also cheaply assert that $J_t$ is indeed zero-dimensional. To check for a solution in $\mathbb{F}_q^n$, one of the following techniques can be applied:

1) Using standard techniques for zero-dimensional ideals, such as order transformation to a lexicographic Gröbner basis followed by back-substitution for extracting solutions in $\mathbb{F}_q^n$. As the ideal is zero-dimensional, efficient algorithms for the order transformation (such as FGLM [13]) can be utilized.
2) Alternatively, we can use the model generation approach presented in [31]. Utilizing the Gröbner basis $G_t$ of zero-dimensional $J_t$, it generates minimal (univariate) polynomials to be factored for solutions in $\mathbb{F}_q$. For zero-dimensional ideals, we are guaranteed to avoid the exponential brute-force case of this approach. We have adopted this method in our implementation.

## V. Experiments

***Implementation Details.*** The theory solver for finite fields in CVC5 [31] uses Gröbner bases as its reasoning backbone, together with CoCoA [1] as an out-of-the-box engine for Gröbner bases calculation. We implemented our approach as an additional (optional) feature of CVC5, extending finite fields reasoning with the following features:

- Implementing the generation of hyperplanes based on a given dimension as presented in Algorithm 1. After adding the hyperplanes, a Gröbner basis calculation is performed. If the resulting Gröbner basis indicates that the ideal is non-empty and zero-dimensional, model construction is attempted, returning a model if found. The resource budget for this check can be specified by the number of tries $T$ or by a time budget for Gröbner basis

calculation used in Algorithm 1. If no satisfiable solution can be found within budget, a regular (complete) attempt to solve the system can be performed (using standard Gröbner basis computation), if configured.
- The dimension of the ideal can either be supplied as a parameter to the problem, or estimated using the Jacobian matrix dimension heuristic, as presented in Section IV-B.

Both features are optional and are designed to be used in addition to the existing solving technique for complicated and high-dimensional systems.

***Benchmarks.*** We evaluate our approach on two established classes of benchmarks from the state-of-the-art in algebraic geometry and computer algebra:

(1) *Mayr–Meyer ideals* $J(n, d)$ [30], [38]: a family of ideals defined over a polynomial ring in $8n + 2$ variables, parameterized by an integer $n \geq 1$ controlling the number of variables and an integer $d \geq 2$ controlling the maximal degree of the generators. The generators of $J(n, d)$ have degree at most $d + 2$, while the (Krull) dimension of the quotient ring is $\dim(R/J(n, d)) = 8n$.

(2) *Randomized determinantal ideals* $I_r(M)$: ideals generated by all $r \times r$ minors of an $m \times n$ matrix $M$ whose entries are random linear forms in a polynomial ring over $N$ variables. For a generic choice of $M$, the (Krull) dimension is given by $\dim(R/I_r(M)) = N - (m - r + 1)(n - r + 1)$ [10].

Both benchmark families are known to be notoriously challenging for Gröbner basis computations, which form the computational backbone of many existing SMT solvers for nonlinear arithmetic over finite fields. In particular, Mayr–Meyer ideals constitute a classical worst-case construction: Gröbner basis computation for this family is known to exhibit doubly exponential complexity in the number of variables in the worst case [30], [38]. Randomized determinantal ideals, while not associated with asymptotic bounds of the same strength, are widely used as practical stress tests and are known to cause significant performance degradation in state-of-the-art Gröbner basis implementations [11], [12]. For both benchmark families, closed-form expressions for the Krull dimension are available as mentioned above.

In addition to the benchmark sets above, we also created a set of random benchmarks with unknown Krull dimension:

(3) *General random ideals* $I(m, n, d)$: ideals generated by a random rectangular system of $m$ polynomials in $n$ variables of degree at most $d$. As we are interested in positive dimensional systems, we did not consider overspecified systems with $m > n$.

To obtain a diverse set of benchmark instances, we vary the parameters for each construction as well as the size of the underlying finite field. Randomness is introduced by uniformly sampling coefficients from the base field.[1]

All our benchmarks were encoded in SMT-LIB using the proposed theory definition for finite fields [21]. We selected these benchmark families to provide a meaningful evaluation of the proposed approach. In particular, our goal is to assess the impact of randomized slicing on classes of problems that are known to be challenging for Gröbner basis computations. In contrast, for many application-oriented benchmarks, such as those considered in [31], Gröbner basis computations are often already efficient in practice. In such settings, the overhead of introducing additional constraints, such as random hyperplanes, may not lead to significant improvements. Therefore, our evaluation focuses on problem classes where dimension reduction is expected to have the greatest impact. This choice reflects the intended use case of our approach: rather than replacing existing techniques, randomized slicing is designed as a complementary method for handling structurally hard and high-dimensional systems.

***Experimental Setup.*** Our experiments were conducted on a 2.5 GHz AMD EPYC 7502 equipped server running Debian. Timeout per instance was set to 5 minutes, with memory limit 16 GB. We kept on running the independent trials for each instance until we either found a solution or exhausted the time limit.

***Research questions.*** Since we are primarily interested in the performance of approaches for satisfiable instances, we focus our evaluation on the following research questions:

- **Effectiveness:** How many additional instances can our randomized approach solve compared to existing complete methods?
- **Efficiency:** What is the average runtime of our randomized approach relative to existing methods?

***Results.*** To evaluate the effectiveness of our approach, we compare the number of instances solved by our randomized slicing method against the baseline CVC5 [4] (without slicing) across all benchmark sets in Table I. We categorize the results into three main groups: instances solved by both solvers (Shared), instances uniquely solved by baseline CVC5 (Base uniq.), and instances uniquely solved by our approach (Ours uniq.). Additionally, we report the number of instances that timed out for both solvers within the specified time limit of 300 seconds.

To evaluate the efficiency of our approach, we present a comparison of runtimes between baseline CVC5 and our approach across all three benchmark sets,[2] as illustrated in Figure 1.

***Jacobian-based dimension estimation.*** We used our general random ideals $I(m, n, d)$ as a testbed for this evaluation, as we don't know a closed form solution for the dimension of these ideals. We observed that for $93.8\%$ of the instances, the Jacobian-based heuristic correctly predicted the true dimension of the ideal, as verified by an exact computation using Gröbner bases.

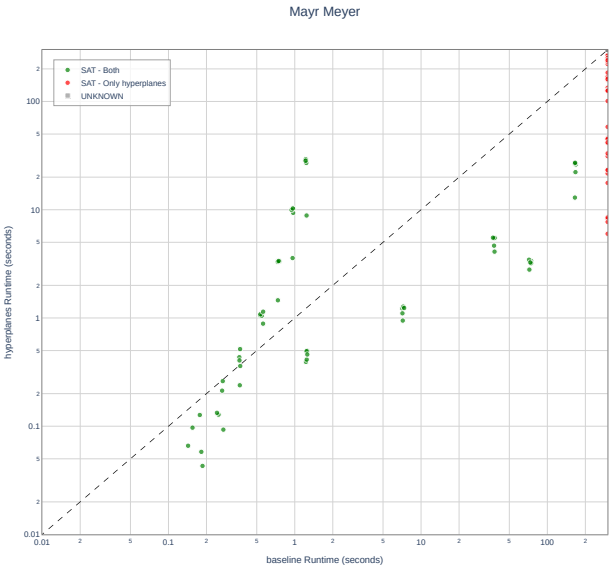
***Discussion.*** The experimental results show that randomized slicing is an effective complementary technique for solving

[1]For more detailed description of the benchmark set see the extended version [16].

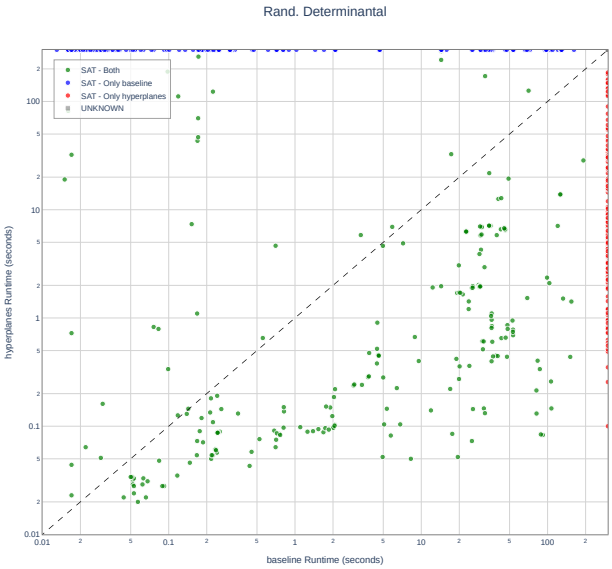
[2]In the extended version [16], we present further experimental results, especially on the performance when running only a limited number of trails.

| Benchmark | Total | Shared | Base uniq. | Ours uniq. | Timeout |
|---|---|---|---|---|---|
| Mayr–Meyer | 420 | 60 | 0 | 44 | 316 |
| Rand. determinantal | 885 | 218 | 223 | 156 | 288 |
| General random | 2400 | 1070 | 426 | 149 | 755 |
| **All benchmarks** | 3705 | 1348 | 649 | 349 | 1359 |

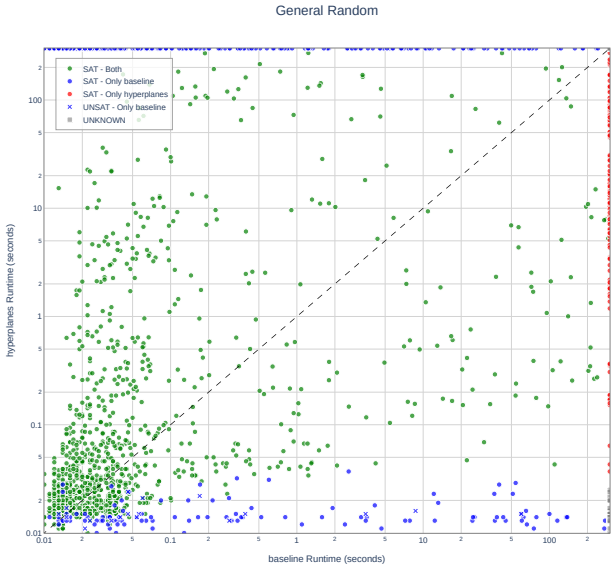
TABLE I: Overall comparison between baseline CVC5 [4] and our approach.




(a) Mayr–Meyer




(b) Rand. determinantal




(c) General random

Fig. 1: Runtime comparison between baseline CVC5 and our approach. Green dots indicate solved by both solvers, blue and red dots are only solved by baseline and hyperplane approach, respectively. Note that blue dots with short runtimes are instances where the Jacobian-based heuristic detects a 0-dim ideal before adding hyperplanes.

nonlinear systems over finite fields. In particular, our approach solves a substantial number of additional instances within the time limit, as highlighted in Table I. These gains are most pronounced for high-dimensional and structurally hard benchmark set of Mayr–Meyer ideals.

At the same time, for instances solved by both solvers, our method exhibits runtime comparable to the baseline, indicating that the overhead introduced by slicing remains moderate in

practice. The improvements are also reflected in the runtime analysis in Figure 1, where our approach enables the solver to reach solutions that are otherwise out of reach for standard Gröbner basis computation.

These results suggest that randomized slicing should be viewed as a complementary strategy, particularly for positive-dimensional systems that are difficult for traditional methods. We also observed that our dimension estimation heuristic is effective for generically generated random systems.

Finally, for certain benchmark instances where Gröbner basis computation is already efficient, the benefit of slicing is limited. In such cases, adding additional linear constraints may introduce overhead without improving performance. This is especially visible for the general random benchmark results where easy benchmarks which can be solved by the baseline instantly are hard for our approach. There are also zero-dimensional benchmarks, correctly characterized by our heuristic, where no hyperplanes can be added. In this case our algorithm stops right away (blue dots in Figure 1c).

## VI. Related Work

SMT reasoning over finite fields has gained attention only recently and remains far less mature than its counterpart for non-linear real arithmetic. An MCSat-based approach for finite fields was proposed in [21], combining a CDCL-style search procedure with theory reasoning tailored to finite field constraints; this line was subsequently implemented in Yices2 [20]. Alternative approaches to SMT reasoning over finite fields are presented in [31], [32]; here, the use of field polynomials are avoided and a splitting technique is introduced to facilitate Gröbner basis computations.

A related line of work studies algebraic decision procedures for finite fields, including quantifier elimination via Gröbner bases. Quantifier-elimination techniques over finite fields using Gröbner bases and field polynomials are introduced in [15]. While such algebraic methods are foundational, directly deploying them within SMT is challenging due to scalability issues, especially for large field orders and when the the solution set has positive dimension.

Mature computer-algebra systems such as Singular [19], CoCoA [1], and Macaulay2 [18] provide highly optimized Gröbner basis implementations over finite fields; however, they are not designed as SMT solvers and do not natively support the broader SMT workflow, nor are they optimized for satisfiability checking. Our implementation builds on the current finite field support in cvc5 [31], [32]. These works develop the core finite field theory solver infrastructure in cvc5 which builds upon calculating Gröbner bases.

On the theoretical side, reducing the dimension by intersecting with a (sufficiently) generic hyperplane is a classical technique in algebraic geometry over characteristic zero; various forms of Bertini's theorem [24] make this intuition precise. In the last few decades, Bertini-type results have also been extended to the finite field setting [36]. However, these theorems are typically formulated in terms of the Zariski topology and therefore do not directly translate to the probabilistic setting needed by our randomized algorithm. Nevertheless, they provide strong motivation for our approach.

Our work is also related in spirit to randomized isolation techniques for Boolean satisfiability, most notably the Valiant–Vazirani reduction for UNIQUE-SAT [42], which adds random linear constraints to isolate satisfying assignments with high probability. Although both approaches use randomness to shrink the solution space, the underlying settings are fundamentally different. Valiant–Vazirani applies to propositional SAT formulas over Boolean domains, whereas our work targets satisfiability modulo theories for non-linear arithmetic over finite fields. More specifically, our method uses random affine hyperplane slices to reduce the geometric dimension of polynomial systems arising in algebraic SMT reasoning, while Valiant–Vazirani uses random XOR constraints to isolate individual Boolean assignments. Consequently, the mathematical tools and analyses required in our setting are distinct from those in the Boolean domain.

## VII. Conclusion

We introduce a randomized algorithm for satisfiability checking the theory of non-linear arithmetic over finite fields. Key to our approach is the use of random affine hyperplanes to reduce the dimension of polynomial constraint systems. Each independent hyperplane typically lowers the dimension by one, and after enough cuts, the ideal becomes with high probability zero-dimensional. This reduction in dimension is crucial as it enables efficient reasoning: expensive symbolic approaches, such as Gröbner basis computation, become far more efficient in the zero-dimensional case.

Our randomized decision procedure is sound and probabilistically slices the solution space until a zero-dimensional system is obtained, after which symbolic methods are used to solve the resulting system. Incompleteness of our method arises only from the random choice of hyperplanes; yet, we show that repeated trials increase the probability of success. Experiments support our work: on benchmarks challenging for symbolic methods, random slicing often transforms instances so Gröbner-basis computations become tractable, yielding significant speedups allowing previously intractable instances to be solved. Overall, combining lightweight randomized dimension reduction with exact symbolic reasoning is effective for finite field verification and related algebraic SMT tasks.


## Acknowledgments

We thank Alex Ozdemir, Jérémy Berthomieu, and Daniela Kaufmann for valuable discussion and input. We further thank Clark Barrett for hosting Thomas Hader at Stanford University while a part of this work was done. We thank the cvc5 team for technical development support.

We acknowledge funding from the ERC Consolidator Grant ARTIST 101002685, the TU Wien SecInt Doctoral College, the FWF SpyCoDe Grant 10.55776/F85, and the WWTF grants [ForSmart Grant ID: 10.47379/ICT22007] and [Promt Grant ID: 10.47379/ICT25017].

## Data availability

The artifact associated with this paper is available at Zenodo under the following DOI: 10.5281/zenodo.22142158.

## References


[1] Abbott, J., Bigatti, A.M., Robbiano, L.: CoCoA: a system for doing Computations in Commutative Algebra. Available at https://sites.google.com/view/cocoa-cocoalib

[2] Atiyah, M.: Introduction to commutative algebra. CRC press (2018)

[3] Barakbayeva, T., Farokhnia, S., Goharshady, A.K., Li, P., Lin, Z.: Improved gas optimization of smart contracts. In: FSEN. pp. 1–10. Springer (2025). https://doi.org/10.1007/978-3-031-87054-5_1, https://doi.org/10.1007/978-3-031-87054-5_1

[4] Barbosa, H., Barrett, C.W., Brain, M., Kremer, G., Lachnitt, H., Mann, M., Mohamed, A., Mohamed, M., Niemetz, A., Nötzli, A., Ozdemir, A., Preiner, M., Reynolds, A., Sheng, Y., Tinelli, C., Zohar, Y.: cvc5: A versatile and industrial-strength SMT solver. In: TACAS. pp. 415–442 (2022)

[5] Buchberger, B.: An Algorithm for Finding the Basis Elements of the Residue Class Ring of a Zero Dimensional Polynomial Ideal. Ph.D. thesis

[6] Collins, G.E.: Quantifier elimination for real closed fields by cylindrical algebraic decompostion. In: Brakhage, H. (ed.) Automata Theory and Formal Languages. Springer Berlin Heidelberg (1975)

[7] Cox, D., Little, J., O'shea, D., Sweedler, M.: Ideals, varieties, and algorithms. Springer (1997)

[8] De Moura, L., Bjørner, N.: Z3: An efficient smt solver. In: International conference on Tools and Algorithms for the Construction and Analysis of Systems. pp. 337–340. Springer (2008)

[9] Dutertre, B.: Yices 2.2. In: CAV. pp. 737–744 (2014)

[10] Eisenbud, D.: Commutative algebra: with a view toward algebraic geometry, vol. 150. Springer Science & Business Media (2013)

[11] Faugere, J.C.: A new efficient algorithm for computing gröbner bases (f4). Journal of pure and applied algebra **139**(1-3), 61–88 (1999)

[12] Faugere, J.C.: A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). In: Proceedings of the 2002 international symposium on Symbolic and algebraic computation. pp. 75–83 (2002)

[13] Faugere, J.C., Gianni, P., Lazard, D., Mora, T.: Efficient computation of zero-dimensional gröbner bases by change of ordering. Journal of Symbolic Computation **16**(4), 329–344 (1993)

[14] Gallian, J.A.: Contemporary Abstract Algebra. Chapman and Hall/CRC (2021)

[15] Gao, S., Platzer, A., Clarke, E.M.: Quantifier elimination over finite fields using gröbner bases. In: International Conference on Algebraic Informatics. pp. 140–157. Springer (2011)

[16] Goharshady, A.K., Hader, T., Kovács, L., Motwani, H.J.: Randomized satisfiability checking for non-linear arithmetic over finite fields (extended version). In: Proceedings of the 26th Conference on Formal Methods in Computer-Aided Design (FMCAD 2026). Graz, Austria (sep 2026), hal-05708451

[17] Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems. In: Providing sound foundations for cryptography: On the work of Shafi Goldwasser and Silvio Micali, pp. 203–225 (2019)

[18] Grayson, D.R., Stillman, M.E.: Macaulay2, a software system for research in algebraic geometry. Available at http://www2.macaulay2.com

[19] Greuel, G.M., Pfister, G., Schönemann, H.: Singular—a computer algebra system for polynomial computations. In: Symbolic computation and automated reasoning, pp. 227–233. AK Peters/CRC Press (2001)

[20] Hader, T., Kaufmann, D., Irfan, A., Graham-Lengrand, S., Kovács, L.: MCSat-based finite field reasoning in the Yices2 SMT solver (short paper). In: International Joint Conference on Automated Reasoning. pp. 386–395. Springer (2024)

[21] Hader, T., Kaufmann, D., Kovács, L.: SMT solving over finite field arithmetic. In: Proceedings of 24th International Conference on Logic. vol. 94, pp. 238–256 (2023)

[22] Hankerson, D., Vanstone, S., Menezes, A.: Guide to elliptic curve cryptography. Springer (2004)

[23] Hartshorne, R.: Algebraic geometry, vol. 52. Springer Science & Business Media (2013)

[24] Kleiman, S.L.: Bertini and his two fundamental theorems. arXiv preprint alg-geom/9704018 (1997)

[25] Kofnov, A., Moosbrugger, M., Stankovic, M., Bartocci, E., Bura, E.: Exact and Approximate Moment Derivation for Probabilistic Loops With Non-Polynomial Assignments. ACM Trans. Model. Comput. Simul. **34**(3), 18:1–18:25 (2024). https://doi.org/10.1145/3641545, https://doi.org/10.1145/3641545

[26] Lakshman, Y.N.: A single exponential bound on the complexity of computing gröbner bases of zero dimensional ideals. In: Effective methods in algebraic geometry. pp. 227–234. Springer (1991)

[27] Lakshman, Y.N., Lazard, D.: On the complexity of zero-dimensional algebraic systems. In: Effective methods in algebraic geometry. pp. 217–225. Springer (1991)

[28] Lang, S., Weil, A.: Number of points of varieties in finite fields. American Journal of Mathematics **76**(4), 819–827 (1954)

[29] MacWilliams, F.J., Sloane, N.J.A.: The theory of error-correcting codes, vol. 16. Elsevier (1977)

[30] Mayr, E.W., Meyer, A.R.: The complexity of the word problems for commutative semigroups and polynomial ideals. Advances in mathematics **46**(3), 305–329 (1982)

[31] Ozdemir, A., Kremer, G., Tinelli, C., Barrett, C.: Satisfiability modulo finite fields. In: International Conference on Computer Aided Verification. pp. 163–186. Springer (2023)

[32] Ozdemir, A., Pailoor, S., Bassa, A., Ferles, K., Barrett, C., Dillig, I.: Split gröbner bases for satisfiability modulo finite fields. In: International Conference on Computer Aided Verification. pp. 3–25. Springer (2024)

[33] Ozdemir, A., Wahby, R.S., Brown, F., Barrett, C.: Bounded verification for finite-field-blasting in a compiler for zero knowledge proofs. Formal Methods in System Design pp. 1–28 (2025)

[34] Passmore, G.O., Jackson, P.B.: Combined decision techniques for the existential theory of the reals. In: International Conference on Intelligent Computer Mathematics. pp. 122–137. Springer (2009)

[35] Platzer, A., Quesel, J., Rümmer, P.: Real World Verification. In: CADE. Springer (2009)

[36] Poonen, B.: Bertini theorems over finite fields. Annals of mathematics pp. 1099–1127 (2004)

[37] Schwartz, J.T.: Fast probabilistic algorithms for verification of polynomial identities. Journal of the ACM (JACM) **27**(4), 701–717 (1980)

[38] Swanson, I.: The minimal components of the mayr-meyer ideals. arXiv preprint math/0209172 (2002)

[39] Szabo, N.: Smart contracts: building blocks for digital markets. EXTROPY: The journal of transhumanist thought **18**(2), 28 (1996)

[40] Tao, T.: The Lang-Weil bound. What's new (blog) (August 31 2012), https://terrytao.wordpress.com/2012/08/31/the-lang-weil-bound/, accessed: February 11, 2026

[41] Valiant, L.G.: The complexity of computing the permanent. Theoretical computer science **8**(2), 189–201 (1979)

[42] Valiant, L.G., Vazirani, V.V.: Np is as easy as detecting unique solutions. In: Proceedings of the seventeenth annual ACM symposium on Theory of computing. pp. 458–463 (1985)

[43] Zippel, R.: Probabilistic algorithms for sparse polynomials. In: International symposium on symbolic and algebraic manipulation. pp. 216–226. Springer (1979)