

SUB-LINEAR ROOT DETECTION AND NEW HARDNESS RESULTS FOR SPARSE POLYNOMIALS OVER FINITE FIELDS *

JINGGUO BI[†], QI CHENG[¶], AND J. MAURICE ROJAS^{||}

Abstract. We present a deterministic $2^{O(t)}q^{\frac{t-2}{t-1}+o(1)}$ algorithm to decide whether a univariate polynomial f , with t monomial terms and degree $< q$, has a root in the finite field \mathbb{F}_q . Our method is the first with complexity *sub-linear* in q when t is fixed. We also prove a structural property for the nonzero roots in \mathbb{F}_q of any t -nomial: The nonzero roots always admit a partition into no more than $2(q-1)^{\frac{t-2}{t-1}}$ cosets, each associated with one of two subgroups $S_1 \subseteq S_2$ of \mathbb{F}_q^* . This can be thought of as a finite field analogue of Descartes' Rule. A corollary of our results is the first deterministic sub-linear algorithm for detecting common degree one factors of k -tuples of t -nomials in $\mathbb{F}_q[x]$ when k and t are fixed.

When t is not fixed we show that, for p prime, detecting roots in \mathbb{F}_p for f is **NP**-hard with respect to **BPP**-reductions. Finally, we prove that if the complexity of root detection is sub-linear (in a refined sense), relative to the *straight-line program encoding*, then **NEXP** $\not\subseteq$ **P/poly**.

Key words. Solvability, sparse polynomial, finite fields, **NP**-hardness, gcd, square-free, discriminant, resultant

AMS subject classifications. 68M07,68Q25,68W40

1. Introduction. The solvability of univariate sparse polynomials is a fundamental problem in computer algebra, and an important precursor to deep questions in polynomial system solving and circuit complexity. Cucker, Koiran, and Smale [18] found a polynomial-time algorithm to find all integer roots of a univariate polynomial f in $\mathbb{Z}[x]$ with exactly t terms, i.e., a *univariate t -nomial*. Shortly afterward, H. W. Lenstra, Jr. [38] gave a polynomial-time algorithm to compute all factors of fixed degree over an algebraic extension of \mathbb{Q} of fixed degree (and thereby all rational roots). Independently, Kaltofen and Koiran [31] and Avendano, Krick, and Sombra [3] extended this to finding bounded-degree factors of sparse polynomials in $\mathbb{Q}[x, y]$ in polynomial-time. Unlike the famous LLL factoring algorithm [37], the complexity of the algorithms from [18, 38, 31, 3] was relative to the *sparse encoding* (cf. Definition 2.1 of Section 2 below) and thus polynomial in $t + \log \deg f$ and the coefficient heights.

Changing the ground field dramatically changes the complexity. For instance, while polynomial-time algorithms are now known for detecting real roots for trinomials in $\mathbb{Z}[x]$ [43, 9], no polynomial-time algorithm is known for tetranomials [5] (counting bit operations). Also, detecting p -adic rational roots for trinomials in $\mathbb{Z}[x]$ was only recently shown to lie in **NP** (for any fixed prime p), as was **NP**-hardness with respect to **ZPP**-reductions for t -nomials when neither t nor p are fixed [2, Thm. 1.4 & Cor. 1.5].

Here, we focus on the complexity of detecting solutions of univariate t -nomials over finite fields.

*A preliminary version of this work appears in the Proceedings of ISSAC 2013 [8]. Jingguo Bi was partially supported by: 973 Program (Grant 2013CB834205), NSF of China under grants No.61502269 & 61133013 & 61272035, and the Open Project Program of the State Key Laboratory of Mathematical Engineering and Advanced Computing No.2014A04. Qi Cheng was partially supported by 973 Program (Grant 2013CB834201) and NSF under grant CCF-1409294. J. Maurice Rojas was partially supported by NSF MCS grant DMS-0915245 and NSF grant CCF-1409020.

[†]Institute for Advanced Study, Tsinghua University, Beijing, China. (jingguobi@mail.tsinghua.edu.cn)

[¶]School of Computer Science, University of Oklahoma, Norman, Oklahoma 73019, USA. (qcheng@cs.ou.edu)

^{||}Department of Mathematics, Texas A&M University, College Station, Texas 77843-3368, USA. (rojas@math.tamu.edu)

1.1. Main Results and Related Work. While deciding the existence of a d^{th} root of an element of the q -element field \mathbb{F}_q is doable in time polynomial in $\log(d) + \log q$ (see, e.g., [4, Thms. 5.6.2 & 5.7.2, pg. 109]), detecting roots for a *trinomial* equation $c_1 + c_2x^{a_2} + c_3x^{a_3} = 0$ with $q - 1 > a_3 > a_2 > 0$ within time sub-linear in q is already a mystery. Around 2003, Erich Kaltofen and David A. Cox independently asked if there exists an algorithm for this problem, with complexity polynomial in $\log q$ [30, 17]. We make progress on a natural extension of this question. In what follows, we use $|S|$ for the cardinality of a set S .

THEOREM 1.1. *Given any univariate t -nomial*

$$f(x) := c_1 + c_2x^{a_2} + c_3x^{a_3} + \cdots + c_tx^{a_t} \in \mathbb{F}_q[x]$$

with degree $< q - 1$, we can decide, within $2^{O(t)}q^{\frac{t-2}{t-1}+o(1)}$ deterministic bit operations, whether f has a root in \mathbb{F}_q . Moreover, letting $\delta := \gcd(q - 1, a_2, \dots, a_t)$ and $\eta := \left(\frac{q-1}{\delta}\right)^{\frac{t-2}{t-1}}$, the entire set of nonzero roots of f in \mathbb{F}_q is a union of at most 2η cosets, each associated to one of two subgroups $S_1 \subseteq S_2$ of \mathbb{F}_q^ , where $|S_1| = \delta$, $|S_2| \geq \delta^{\frac{t-2}{t-1}}(q-1)^{\frac{1}{t-1}}$, and $|S_2|$ can be determined in time $2^{O(t)}(\log q)^{O(1)}$.*

We prove Theorem 1.1 in Section 3.1. The degree assumption is natural since $x^q = x$ in $\mathbb{F}_q[x]$. Note also that deciding whether an f as above has a root in \mathbb{F}_q via brute-force search takes $q^{1+o(1)}$ bit operations, assuming t is fixed. If we pick a_2, \dots, a_t uniformly randomly in $\{-M, \dots, M\}$ then, as $M \rightarrow \infty$, the probability that $\gcd(a_2, \dots, a_t) = 1$ approaches $1/\zeta(t-1)$ (see, e.g., [16]). The latter quantity increases from $\frac{6}{\pi^2} \approx 0.6079$ to 1 as t goes from 3 to ∞ . Our theorem thus implies that, with “high” probability over the inputs, the nonzero roots of a sparse polynomial over a finite field can be divided into two components: One component consisting of no more than q^c (for some $c < 1$) isolated roots, and the other component consisting of q^c cosets of a (potentially large) subgroup of \mathbb{F}_q^* . Put another way, if the number of nonzero roots of a univariate t -nomial in \mathbb{F}_q^* is larger than $2(q-1)^{\frac{t-2}{t-1}}$, then the roots must exhibit a strong multiplicative structure.

The classic *Descartes’ Rule* [46] implies that the number of distinct real roots of a real univariate t -nomial is at most $2t - 1$, regardless of the degree. At first glance, one would think that the polynomial $x^{q-1} - 1 \in \mathbb{F}_q[x]$ immediately renders a finite field analogue impossible. On the other hand, note that the nonzero roots of any binomial form a coset of a subgroup of \mathbb{F}_q^* . Our first main result indicates that, over a finite field, the number of *cosets needed to cover* the set of nonzero roots of a sparse polynomial f is much smaller than the degree of f . We thus obtain a finite field analogue of Descartes’ Rule. We consider the new idea of counting by cosets as one of the main contributions of this paper. More to the point, Theorem 1.1 provides new structural and algorithmic information, complementing an earlier finite field analogue of Descartes’ Rule [11, Lemma 7]. Theorem 1.1 can also be thought of as a refined, positive characteristic analogue of results of Tao and Meshulam [47, 39] bounding the number of complex roots of unity at which a sparse polynomial can vanish (a.k.a. uncertainty inequalities over finite groups).

REMARK 1.2. *There is recent evidence that our upper bounds are not far from optimality, for a large family of fields: For any prime p and $q = p^k$ with k a multiple of t , there are univariate t -nomials in $\mathbb{F}_q[x]$ with $1 + q^{1/t} + \cdots + q^{(t-2)/t}$ nonzero roots in \mathbb{F}_q (and $\delta = 1$) [14]. In particular, this implies that the total number of cosets from our first main theorem can be as large as $1 + q^{1/t} + \cdots + q^{(t-2)/t}$ when $\delta = 1$. See also [35] for the optimal upper bound of \sqrt{q} roots when $(\delta, q, t) = (1, p^k, 3)$ with $p \geq 3$ and k*

even. The nature of optimal upper bounds for trinomials over prime fields appears to be more subtle: The existence of a constant $\gamma > 0$, an increasing sequence of primes $(p_i)_{i \in \mathbb{N}}$, and a sequence of trinomials $(f_i)_{i \in \mathbb{N}}$, with each $f_i \in \mathbb{F}_{p_i}[x]$ having $\delta = 1$ and at least $\gamma \log p_i$ roots in \mathbb{F}_{p_i} for all $i \in \mathbb{N}$, is unknown [14, 35]. \diamond

Since detecting roots over \mathbb{F}_q is the same as detecting linear factors of polynomials in $\mathbb{F}_q[x]$, it is natural to ask about the complexity of factoring sparse polynomials over $\mathbb{F}_q[x]$. The asymptotically fastest randomized algorithm for factoring arbitrary $f \in \mathbb{F}_q[x]$ of degree d uses $O(d^{1.5} + d^{1+o(1)} \log q)$ bit operations [34], but no complexity bound polynomial in $t + \log(d) + \log q$ is known. (See [7, 12, 25, 32, 48] for some important milestones, and [24, 30, 22] for an extensive survey on factoring.) However, to detect roots in \mathbb{F}_q , we don't need the full power of factoring: We need only decide whether $\gcd(x^q - x, f(x))$ has positive degree. Indeed, a consequence of our first main result is a speed-up for a variant of the latter decision problem.

COROLLARY 1.3. *Given any univariate t -nomials $f_1, \dots, f_k \in \mathbb{F}_q[x]$, we can decide if f_1, \dots, f_k have a degree one factor in $\mathbb{F}_q[x]$ in common via a deterministic algorithm with complexity $2^{O(kt-k)} q^{\frac{kt-k-1}{kt-k} + o(1)}$.*

Corollary 1.3 (proved in Section 3.2) appears to give the first sub-linear algorithm for detecting roots of k -tuples of univariate t -nomials for k and t fixed.

REMARK 1.4. *It is important to note that the $k = 2$ case is not the same as deciding whether the gcd of two general polynomials has positive degree: The latter problem is the same as detecting common factors of arbitrary degree, or degree one factors over an extension field. Finding an algorithm for the latter problem with complexity sub-linear in q is already an open problem for $k = 2$ and $t \geq 3$: See [20], and Theorem 1.6 and Remark 1.8 below. \diamond*

One reason why it is challenging to attain complexity sub-linear in q is that detecting roots in \mathbb{F}_q for t -nomials is **NP**-hard when t is not fixed, even restricting to one variable and prime q .

THEOREM 1.5. *Suppose that, for any input (f, p) with p a prime and $f \in \mathbb{F}_p[x]$ a t -nomial of degree $< p$, one could decide whether f has a root in \mathbb{F}_p within **BPP**, using $t + \log p$ as the underlying input size. Then $\mathbf{NP} \subseteq \mathbf{BPP}$.*

We prove Theorem 1.5 in Section 4.1. The least (fixed) n making root detection in \mathbb{F}_p^n be **NP**-hard for polynomials in $\bigcup_{p \text{ prime}} \mathbb{F}_p[x_1, \dots, x_n]$ (relative to the sparse encoding)

appears to have been unknown. Theorem 1.5 thus settles this problem, provided we allow (**BPP**-) randomized reductions. Theorem 1.5 also complements an earlier result of Kipnis and Shamir proving **NP**-hardness for detecting roots of univariate sparse polynomials over fields of the form \mathbb{F}_{2^t} [36]. Furthermore, Theorem 1.5 improves another recent **NP**-hardness result where the underlying input size was instead the (smaller) straight-line program complexity [15].

Let $\overline{\mathbb{F}}_q$ denote the algebraic closure of \mathbb{F}_q . A consequence of our last complexity lower bound is the hardness of detecting *degenerate* roots over \mathbb{F}_p and $\overline{\mathbb{F}}_p$:

THEOREM 1.6. *Consider the following two problems, each with input (f, p) where p is a prime and $f \in \mathbb{F}_p[x]$ is a t -nomial of degree $< p$.*

1. *Decide whether f is divisible by the square of a degree one polynomial in $\mathbb{F}_p[x]$.*
2. *Decide whether f is divisible by the square of a degree one polynomial in $\overline{\mathbb{F}}_p[x]$.*

*Then, using $t + \log p$ as the underlying input size, each of these problems is **NP**-hard with respect to **BPP**-reductions.*

We prove Theorem 1.6 in Section 4.2. The **NP**-hardness of both problems had been previously unknown. Theorem 1.6 thus improves [33, Cor. 2] where **NP**-hardness

(with respect to **BPP**-reductions) was proved for the *harder* variant of Problem (2) where one allows f in the larger ring $\overline{\mathbb{F}}_p[x]$.

REMARK 1.7. *Note that detecting a degenerate root for f is the same as detecting a common degree one factor of f and $\frac{\partial f}{\partial x}$, at least when $\deg f$ is less than the characteristic of the field. So an immediate consequence of Theorem 1.6 is that detecting common degree one factors in $\mathbb{F}_p[x]$ (resp. $\overline{\mathbb{F}}_p[x]$) for pairs of polynomials in $\mathbb{F}_p[x]$ is **NP**-hard with respect to **BPP**-reductions. We thus also strengthen earlier work proving similar complexity lower bounds for detecting common degree one factors in $\mathbb{F}_q[x]$ (resp. $\overline{\mathbb{F}}_q[x]$) [23, Thm. 4.11]. \diamond*

REMARK 1.8. *It should be noted that Problem (2) in Theorem 1.6 is equivalent to deciding the vanishing of univariate \mathcal{A} -discriminants (see [26, Ch. 12, pp. 403–408] and Definitions 2.6 and 2.8 of Section 2.2 below). While the trinomial case of Problem (2) can be done in **P** (see [2, Lemma 5.3]), we are unaware of any other speed-ups for fixed t . In particular, it follows immediately from Theorem 1.6 that deciding the vanishing of univariate resultants (see, e.g., [26, Ch. 12, Sec. 1, pp. 397–402] and Definition 2.6 of Section 2.2 below), of polynomials in $\mathbb{F}_p[x]$, is also **NP**-hard with respect to **BPP**-reductions. \diamond*

Our final main result is a complexity separation depending on a weak tractability assumption for detecting roots of univariate polynomials given as *straight-line programs (SLPs)*.

DEFINITION 1.9. *A straight-line program for a polynomial $f(x_1, \dots, x_d)$ over a field F is a sequence of assignments. The i -th assignment is*

$$v_i \leftarrow v_m \circ v_n, \text{ or } v_i \leftarrow c_i, \text{ or } v_i \leftarrow x_k$$

where $m < i, n < i$, $\circ \in \{+, -, \times\}$, and c_i is any constant in F . If we run the program, the last assignment outputs f . The length of the program is the number of assignments.

THEOREM 1.10. *Suppose that, given any straight-line program of size L representing a polynomial $f \in \mathbb{F}_{2^\ell}[x]$, we could decide if f has a root in \mathbb{F}_{2^ℓ} within time $L^{O(1)}2^{\ell - \omega(\log \ell)}$. Then **NEXP** $\not\subseteq$ **P/poly**.*

We prove Theorem 1.10 in Section 4.3. One should recall that **NEXP** \subseteq **P/poly** \iff **NEXP** = **MA** [28]. So the conditional assertion of our last theorem indeed implies a new separation of complexity classes. It may actually be the case that there is no algorithm for detecting roots in \mathbb{F}_{2^ℓ} better than brute-force search. Such a result would be in line with the Exponential Time Hypothesis [29] and the widely-held belief in the cryptographic community that the only way to break a well-designed block cipher is by exhaustive search.

1.2. Highlights of Main Techniques. Let e be a positive integer such that $\gcd(e, q-1) = 1$. If we replace x by x^e in

$$f(x) = c_1 + c_2x^{a_2} + c_3x^{a_3} + \dots + c_tx^{a_t} \in \mathbb{F}_q[x],$$

then we obtain

$$f(x^e) = c_1 + c_2x^{ea_2} + c_3x^{ea_3} + \dots + c_tx^{ea_t}.$$

These two polynomials have the same number of roots in \mathbb{F}_q since the map from \mathbb{F}_q to \mathbb{F}_q given by $x \mapsto x^e$ is one-to-one. Now suppose that $(m_2, m_3, \dots, m_t) \in \mathbb{Z}^{t-1}$ satisfies $m_2 \equiv ea_2, \dots, m_t \equiv ea_t \pmod{q-1}$. Then f has a root in \mathbb{F}_q iff the polynomial

$$c_1 + c_2x^{m_2} + c_3x^{m_3} + \dots + c_tx^{m_t}$$

has a root in \mathbb{F}_q . The key new advance needed to attain our speed-ups is a method employing recent fast algorithms for the *Shortest Vector Problem (SVP)* in ℓ_∞ (see

[19]). In particular, our method finds a suitable e that lowers the degree of any sparse polynomial in $\mathbb{F}_q[x]$ to a value sub-linear in q while still preserving solvability over \mathbb{F}_q .

LEMMA 1.11. *Given integers a_1, \dots, a_t, N satisfying $0 < a_1 < \dots < a_t < N$ and $\gcd(N, a_1, \dots, a_t) = 1$, one can find, within $2^{O(t)}(t \log N)^{O(1)}$ bit operations, an integer e with the following property for all $i \in \{1, \dots, t\}$: If $m_i \in \{-\lfloor N/2 \rfloor, \dots, \lfloor N/2 \rfloor\}$ is the unique integer congruent to $ea_i \pmod N$ then $|m_i| \leq N^{1-t^{-1}}$.*

We prove this lemma in Section 2.1. The lemma can be applied to the exponents of a general sparse polynomial to yield Theorem 1.1 in Section 3.1, after overcoming two potential difficulties: One can sometimes have $\gcd(q-1, a_1, \dots, a_t) > 1$ or $\gcd(e, q-1) > 1$.

Recall that any logical disjunction of one of the following forms:

$$(\diamond) \quad y_i \vee y_j \vee y_k, \quad \neg y_i \vee y_j \vee y_k, \quad \neg y_i \vee \neg y_j \vee y_k, \quad \neg y_i \vee \neg y_j \vee \neg y_k,$$

with $i, j, k \in \{1, \dots, n\}$ not necessarily distinct, is a (3-SAT) clause. In particular, at the possible expense of additional variables, any logical formula from propositional calculus is equivalent to a conjunction of 3-SAT clauses. A formula that is a conjunction of disjunctions of (possibly negated) variables is said to be in *conjunctive normal form*, and a *satisfying assignment* for a logical formula $B(y_1, \dots, y_n)$ is an assignment of values from $\{0, 1\}$ to the variables y_1, \dots, y_n which makes the equality $B(y_1, \dots, y_n) = 1$ true.¹ 3CNFSAT is the well-known NP-complete problem of deciding the existence of a satisfying assignment, for formulas in conjunctive normal form, where all clauses are in the form (\diamond) [21, 1].

A key construction behind the proofs of Theorems 1.5 and 1.6 in Section 4 is a randomized reduction from 3CNFSAT to detecting roots of univariate polynomial systems over finite fields. In particular, the finite fields arising in this reduction have cardinality coming from a very particular family of prime numbers. (See Definition 2.1 from Section 2 for our definition of input size.)

THEOREM 1.12. [2, Secs. 6.2–6.3]² *There is a (Las Vegas) randomized polynomial-time algorithm that, given any 3CNFSAT instance $B(y_1, \dots, y_n)$ in $n \geq 4$ variables with k clauses, produces positive integers c, p_1, \dots, p_n and polynomials $f_1, \dots, f_k \in (\mathbb{Z}/(1 + \prod_{j=1}^n p_j) \mathbb{Z})[x]$ with the following properties:*

1. $c \geq 11$ and $\log(c \prod_{j=1}^n p_j) = n^{O(1)}$.
2. p_1, \dots, p_n is an increasing sequence of primes and $p := 1 + c \prod_{j=1}^n p_j$ is prime.
3. For each i , f_i is monic, $f_i(0) \neq 0$, $\deg f_i < \prod_{j=1}^n p_j$, and $\text{size}(f_i) = n^{O(1)}$.
4. For each i , f_i has exactly $\deg f_i$ distinct roots in \mathbb{F}_p .
5. B has a satisfying assignment if and only if the system $f_1(x) = \dots = f_k(x) = 0$ has a solution in \mathbb{F}_p . ■

Theorem 1.12 is based on an earlier reduction of Plaisted involving complex roots of unity [41, Sec. 3, pp. 127–129].

We now review some additional background necessary for our proofs.

2. Background. Our main notion of input size essentially reduces to how long it takes to write down monomial term expansions, a.k.a. the *sparse encoding*.

DEFINITION 2.1. *For any polynomial $f \in \mathbb{Z}[x_1, \dots, x_n]$ expressed in the form $f(x) = \sum_{i=1}^t c_i x_1^{a_{1,i}} \dots x_n^{a_{n,i}}$, with $a_{j,i} > 0$ for all j, i , we define*

¹We respectively identify 0 and 1 with “False” and “True”.

²[2] in fact contains a version of Theorem 1.12 with $c \geq 2$, but $c \geq 11$ can be attained by a trivial modification of the proof there.

$$\text{size}(f) := \sum_{i=1}^t \log_2 [(2 + |c_i|)(2 + a_{1,i}) \cdots (2 + a_{n,i})].$$

Also, when $F := (f_1, \dots, f_k)$, we define $\text{size}(F) := \sum_{i=1}^k \text{size}(f_i)$. \diamond

The definition above is also sometimes referred to as the *sparse size* of a polynomial system. Note that $\text{size}(c) = O(\log |c|)$ for any integer c .

A fact we'll need is that systems of univariate polynomial equations can, at the expense of some randomization, be reduced to *pairs* of univariate equations. (See [27] for a multivariate version.)

LEMMA 2.2. *Given any prime power q and polynomials $f_1, \dots, f_k \in \mathbb{F}_q[x] \setminus \{0\}$, let $Z(f_1, \dots, f_k)$ denote the set of solutions of $f_1 = \dots = f_k = 0$ in $\overline{\mathbb{F}}_q$. Also let $d_1 := \deg f_1$. Then at least a fraction of $1 - \frac{d_1}{q}$ of the $(u_2, \dots, u_k) \in \mathbb{F}_q^{k-1}$ satisfy $Z(f_1, \dots, f_k) = Z(f_1, u_2 f_2 + \dots + u_k f_k)$.*

Proof: Let $\{\zeta_1, \dots, \zeta_r\} := Z(f_1) \setminus Z(f_2, \dots, f_k)$, with the ζ_i pair-wise distinct. Note that $r \leq d_1$. If $r = 0$ then we obtain $Z(f_1) \subseteq Z(f_2, \dots, f_k)$ and thus $Z(f_1, \dots, f_k) = Z(f_1, u_2 f_2 + \dots + u_k f_k) = Z(f_1)$ for any choice of u_i in \mathbb{F}_q . So we may assume $r \geq 1$.

We then observe that, by definition, for any $i \in \{1, \dots, r\}$, we must have $f_{j_i}(\zeta_i) \neq 0$ for some $j_i \in \{2, \dots, k\}$. In particular, the polynomial

$$L(u) := \prod_{i=1}^r (u_{j_i} f_{j_i}(\zeta_i) + \dots + u_k f_k(\zeta_i)) \in \overline{\mathbb{F}}_q[u_2, \dots, u_k]$$

has degree r , and is thus not identically 0. Note also that the statement of the lemma is vacuous when $d_1 \geq q$, so we may assume $d_1 < q$. By the classical *Schwartz-Zippel Lemma* [44, 50], we then obtain that L vanishes at no more than $r q^{k-2}$ choices of $u \in \mathbb{F}_q^{k-1}$. Since $Z(f_2, \dots, f_k) \subseteq Z(u_2 f_2 + \dots + u_k f_k)$ for any choice of u , we thus obtain $L(u) \neq 0 \implies Z(f_1, \dots, f_k) = Z(f_1, u_2 f_2 + \dots + u_k f_k)$, and we are done. \blacksquare

Let us now observe the following complexity bound for root detection for (not necessarily sparse) polynomials over finite fields.

PROPOSITION 2.3. *Given any polynomial $f \in \mathbb{F}_q[x]$ of degree d and $N|(q-1)$, we can decide within $d^{1+o(1)}(\log q)^{2+o(1)}$ deterministic bit operations whether f has a root in the order N subgroup of \mathbb{F}_q^* . \blacksquare*

Since detecting roots for f as above is the same as deciding whether $\gcd(x^N - 1, f(x))$ has positive degree, the complexity bound above can be attained as follows: Compute $r(x) := x^N \bmod f(x)$ via repeated squaring [4, Thm. 5.4.1, pg. 103], and then compute $\gcd(r(x) - 1, f(x))$ in time $d^{1+o(1)}(\log q)^{1+o(1)}$ via the Knuth-Schönhage algorithm [10, Ch. 3].

2.1. Geometry of Numbers for Speed-Ups. For any linearly independent vectors $\mathbf{b}_1, \dots, \mathbf{b}_d \in \mathbb{R}^m$, we call the set $\mathcal{L}(\mathbf{b}_1, \dots, \mathbf{b}_d) = \left\{ \sum_{i=1}^d x_i \mathbf{b}_i \mid x_i \in \mathbb{Z} \right\}$ a *lattice* in \mathbb{R}^m . The integers d and m are respectively called the *rank* and *dimension* of the lattice. Any lattice can be conveniently represented by a $d \times m$ matrix \mathbf{B} with rows $\mathbf{b}_1, \dots, \mathbf{b}_d$. The determinant $\det(\mathcal{L})$ of the lattice \mathcal{L} is the volume of the d -dimensional parallelepiped spanned by the origin and the vectors of any \mathbb{Z} -basis for \mathcal{L} . If an explicit basis of \mathcal{L} is known, the determinant of the lattice \mathcal{L} can then be computed as $\det(\mathcal{L}) = \det(\mathcal{L}(\mathbf{B})) = \sqrt{\det(\mathbf{B}\mathbf{B}^\top)}$.

For any vector $\mathbf{u} = (u_1, \dots, u_m) \in \mathbb{R}^m$ we let $\|\mathbf{u}\|_p := \sqrt[p]{|u_1|^p + \dots + |u_m|^p}$ denote its ℓ_p norm. (We also set $\|\mathbf{u}\|_\infty := \max_i |u_i|$.) Perhaps the most famous computational problem on lattices is the (*exact*) *Shortest Vector Problem (SVP)*: Given a basis of a lattice \mathcal{L} , find a non-zero vector $\mathbf{u} \in \mathcal{L}$, such that $\|\mathbf{v}\|_p \geq \|\mathbf{u}\|_p$ for any vector $\mathbf{v} \in \mathcal{L} \setminus \mathbf{0}$. To obtain an upper bound on the ℓ_p norm of a shortest vector in a lattice \mathcal{L} , one usually use Minkowski's Convex Body Theorem (see, e.g., [13]):

MINKOWSKI'S CONVEX BODY THEOREM. *Let \mathcal{L} be a full-rank lattice in \mathbb{R}^n . Let C be a measurable subset of \mathbb{R}^n with C convex, centrally symmetric, and of volume strictly greater than $2^n \det(\mathcal{L})$. Then C contains at least one point in $\mathcal{L} \setminus \{\mathbf{0}\}$. ■*

As a corollary of the Convex Body Theorem we can get an upper bound on the ℓ_∞ norm of a shortest vector in $\mathcal{L} \setminus \{\mathbf{0}\}$.

THEOREM 2.4. *Any lattice \mathcal{L} of rank d contains a vector \mathbf{v} with $0 < \|\mathbf{v}\|_\infty \leq \det(\mathcal{L})^{1/d}$.*

Proof: Let S_c be the hypercube $[-c, c]^d$ and assume $c > \det(\mathcal{L})^{1/d}$. Note that S is a measurable subset of \mathbb{R}^d that is convex and symmetric with respect to 0. The volume of S_c is clearly strictly greater than $2^d \det(\mathcal{L})$. So by the Convex Body Theorem, $C \cap \mathcal{L} \setminus \{\mathbf{0}\}$ is non-empty and, by construction, every point of $C \cap \mathcal{L}$ has ℓ_∞ norm at most c . Since \mathcal{L} is a closed set, and $c > \det(\mathcal{L})^{1/d}$ is arbitrary, this means that $S_{\det(\mathcal{L})^{1/d}} \cap \mathcal{L} \setminus \{\mathbf{0}\}$ must be non-empty as well. So we are done. ■

Given a lattice with rank d , most lattice reduction algorithms, such as the celebrated *LLL algorithm* [37], define shortest lattice vectors in terms of the ℓ_2 norm. (See [40] for a survey of other SVP algorithms.) An algorithm with arithmetic complexity $d^{O(1)}2^{O(d)}$, proposed in [19] by Dadush et. al., is currently the fastest deterministic algorithm for solving SVP relative to the ℓ_∞ norm.

Let us now prepare for our degree-lowering tricks. First, we construct the lattice \mathcal{L} spanned by the rows of matrix \mathbf{B} , where

$$(\star\star) \quad \mathbf{B} = \begin{bmatrix} a_1 & a_2 & \cdots & a_t \\ N & 0 & \cdots & 0 \\ 0 & N & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & N \end{bmatrix}$$

Letting $\mathbf{v} := (m_1, \dots, m_t)$ be the shortest vector of the lattice \mathcal{L} , there then clearly exists an integer e such that $ea_1 \equiv m_1, \dots, ea_t \equiv m_t \pmod{N}$. (In fact, e is merely the coefficient of (a_1, \dots, a_t) in the underlying linear combination defining \mathbf{v} .) Most importantly, the factorization of $\det(\mathcal{L})$ is highly constrained when the a_i are relatively prime.

LEMMA 2.5. *If $\gcd(N, a_1, \dots, a_t) = 1$ then $\det(\mathcal{L}) = N^{t-1}$.*

Since no explicit basis of \mathcal{L} is known, we will calculate the determinant by duality: The dual of a lattice \mathcal{L} in \mathbb{R}^n , denoted \mathcal{L}^* , is the lattice given by the set of all vectors $\mathbf{y} \in \mathbb{R}^n$ with inner product $\langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{Z}$ for all vectors $\mathbf{x} \in \mathcal{L}$. We have $\det(\mathcal{L}^*) = 1/\det(\mathcal{L})$. In addition, if \mathcal{L}_1 and \mathcal{L}_2 are two lattices in \mathbb{R}^m with the same dimension and $\mathcal{L}_1 \subseteq \mathcal{L}_2$, then $\mathcal{L}_2/\mathcal{L}_1$ is a finite group of order denoted by $[\mathcal{L}_2 : \mathcal{L}_1]$ which satisfies $\det(\mathcal{L}_1) = \det(\mathcal{L}_2)[\mathcal{L}_2 : \mathcal{L}_1]$.

Proof of Lemma 2.5: Define the lattice

$$\mathcal{L}^\perp = \left\{ \mathbf{x} = (x_1, \dots, x_t) \in \mathbb{Z}^t \mid \sum_{i=1}^t x_i a_i \equiv 0 \pmod{N} \right\}.$$

It is easily checked that $\mathcal{L}^\perp = N\mathcal{L}^*$. Note that \mathcal{L}^\perp and \mathcal{L} are dual to each other, up to normalization. Hence $\det(\mathcal{L}) \times \det(\mathcal{L}^\perp) = N^t$.

We will now prove $\det(\mathcal{L}^\perp) = N$. Since \mathcal{L}^\perp contains all the vectors of the canonical basis of \mathbb{Z}^t multiplied by N , the dimension of \mathcal{L}^\perp is t . It is easily checked that \mathcal{L}^\perp is a subgroup of \mathbb{Z}^t . It then follows that $\det \mathcal{L}^\perp = [\mathbb{Z}^t : \mathcal{L}^\perp]$. Furthermore, the definition of \mathcal{L}^\perp clearly implies that $[\mathbb{Z}^t : \mathcal{L}^\perp] = N/\gcd(N, a_1, a_2, \dots, a_t)$. Hence $\det(\mathcal{L}^\perp) = N/\gcd(N, a_1, a_2, \dots, a_t) = N$ and we are done. ■

We are now ready to prove Lemma 1.11.

Proof of Lemma 1.11: From Lemma 2.5 and Theorem 2.4, there exists a shortest

vector \mathbf{v} of \mathcal{L} satisfying $\|\mathbf{v}\|_\infty \leq N^{1-t^{-1}}$. By invoking the exact SVP algorithm from [19] we can then find the shortest vector \mathbf{v} in time $2^{O(t)}(t \log N)^{O(1)}$. Let $\mathbf{v} := (m_1, \dots, m_t)$. Clearly, by shortness, we may assume $|m_i| \leq N/2$ for all $i \in \{1, \dots, t\}$. (Otherwise, we would be able to reduce m_i in absolute value by subtracting a suitable row of the matrix \mathbf{B} from \mathbf{v} .) Also, by construction, there is an e such that $ea_i \equiv m_i \pmod{N}$ for all $i \in \{1, \dots, t\}$. ■

2.2. Resultants, \mathcal{A} -discriminants, and Square-freeness. Let us first recall the classical univariate resultant.

DEFINITION 2.6. (See, e.g., [26, Ch. 12, Sec. 1, pp. 397–402].) Suppose $f(x) = a_0 + \dots + a_d x^d$ and $g(x) = b_0 + \dots + b_{d'} x^{d'}$ are polynomials with indeterminate coefficients. We define their Sylvester matrix to be the $(d + d') \times (d + d')$ matrix

$$\mathcal{S}_{(d,d')}(f,g) := \left[\begin{array}{cccccccc} a_0 & a_1 & \cdots & a_{d-1} & a_d & 0 & \cdots & 0 \\ & a_0 & \ddots & a_{d-2} & a_{d-1} & a_d & \ddots & 0 \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & a_0 & a_1 & \cdots & a_{d-1} & a_d \\ b_0 & b_1 & \cdots & b_{d'-1} & b_{d'} & 0 & \cdots & 0 \\ & b_0 & \ddots & b_{d'-2} & b_{d'-1} & b_{d'} & \ddots & 0 \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & b_0 & b_1 & \cdots & b_{d'-1} & b_{d'} \end{array} \right] \left. \begin{array}{l} \vphantom{\mathcal{S}_{(d,d')}(f,g)} \\ \vphantom{\mathcal{S}_{(d,d')}(f,g)} \\ \vphantom{\mathcal{S}_{(d,d')}(f,g)} \\ \vphantom{\mathcal{S}_{(d,d')}(f,g)} \\ \vphantom{\mathcal{S}_{(d,d')}(f,g)} \\ \vphantom{\mathcal{S}_{(d,d')}(f,g)} \\ \vphantom{\mathcal{S}_{(d,d')}(f,g)} \\ \vphantom{\mathcal{S}_{(d,d')}(f,g)} \end{array} \right\} \begin{array}{l} d' \text{ rows} \\ \\ \\ d \text{ rows} \end{array}$$

and their Sylvester resultant to be

$$\text{Res}_{(d,d')}(f,g) := \det \mathcal{S}_{(d,d')}(f,g). \diamond$$

LEMMA 2.7. Following the notation of Definition 2.6, assume $f, g \in K[x]$ for some field K , and that a_d and $b_{d'}$ are not both 0. Then $f = g = 0$ has a root in the algebraic closure of K if and only if $\text{Res}_{(d,d')}(f,g) = 0$. More precisely, we have $\text{Res}_{(d,d')}(f,g) = a_d^{d'} \prod_{f(\zeta)=0} g(\zeta)$, where the product counts multiplicity. ■

The lemma is classical: See, e.g., [26, Ch. 12, Sec. 1, pp. 397–402], [42, pg. 9], and [6, Thm. 4.16, pg. 107] for a more modern treatment.

We may now define a refinement of the classical *discriminant*.

DEFINITION 2.8. (See also [26, Ch. 12, pp. 403–408].) Let $\mathcal{A} := \{a_1, \dots, a_t\} \subset \mathbb{N} \cup \{0\}$ and $f(x) := \sum_{i=1}^t c_i x^{a_i}$, where $0 \leq a_1 < \dots < a_t$ and the c_i are indeterminates. We then define the \mathcal{A} -discriminant of f , $\Delta_{\mathcal{A}}(f)$, to be

$$\text{Res}_{(\bar{a}_1, \bar{a}_t - \bar{a}_2)} \left(\bar{f}, \frac{\partial \bar{f}}{\partial x} / x^{\bar{a}_2 - 1} \right) / c_t^{\bar{a}_t - \bar{a}_t - 1},$$

where $\bar{a}_i := (a_i - a_1)/g$ for all i , $\bar{f}(x) := \sum_{i=1}^t c_i x^{\bar{a}_i}$, and $g := \gcd(a_2 - a_1, \dots, a_t - a_1)$. \diamond

REMARK 2.9. Note that when $\mathcal{A} = \{0, \dots, d\}$ we have $\Delta_{\mathcal{A}}(f) = \text{Res}_{(d,d-1)}(f, f')/c_d$, i.e., for dense polynomials, the \mathcal{A} -discriminant agrees with the classical discriminant. \diamond

LEMMA 2.10. Suppose p is any prime and $f, g \in \mathbb{F}_p[x]$ are relatively prime polynomials satisfying $f(0)g(0) \neq 0$, $d := \deg g \geq \deg f$, and $p > d$. Then the polynomial $f + ag$ is square-free for at least a fraction of $1 - \frac{2d-1}{p}$ of the $a \in \mathbb{F}_p$.

Proof: For $2d-1 \geq p$ the lemma is vacuous, so let us assume $2d-1 < p$. Note also that the polynomial $f + ag$ is irreducible in $\mathbb{F}_p[x, a]$, since f and g have no common factors in $\mathbb{F}_p[x]$. The splitting field $L \subsetneq \overline{\mathbb{F}_p(a)}$ of $f(x) + ag(x)$ must have degree $[L : \mathbb{F}_p(a)]$ dividing $(\deg f)!$. Since $\deg f \leq d < p$, p can not divide $[L : \mathbb{F}_p(a)]$ and thus L is a separable extension of $\mathbb{F}_p(a)$, i.e., $f + ag$ has no degenerate roots in $\overline{\mathbb{F}_p(a)}$. So the

classical discriminant of $f + ag$ (where the coefficients are considered as polynomials in a) is a polynomial in a that is not identically zero. Furthermore, from Definition 2.6, $\text{Res}_{(d,d-1)}(f + ag, f' + ag') \in \mathbb{F}_p[a]$ has degree at most $d + d - 1 = 2d - 1$. So by Lemma 2.2, the classical discriminant of $f + ag$ is non-zero for at least $1 - \frac{2d-1}{p}$ of the $a \in \mathbb{F}_p$. Thanks to Lemma 2.7, we thus obtain that $f + ag$ is square-free for at least a fraction of $1 - \frac{2d-1}{p}$ of the $a \in \mathbb{F}_p$. ■

A stronger assertion, satisfied on a much smaller set of a , was observed earlier in the proof of Theorem 1 of [33]. For our purposes, easily finding an a with $f + ag$ square-free will be crucial.

3. Faster Root Detection: Proving Theorem 1.1 and Corollary 1.3 .

3.1. Proving Theorem 1.1. Before proving Theorem 1.1, let us first prove a result that will in fact enable sub-linear root detection in *arbitrary* subgroups of \mathbb{F}_q^* .

LEMMA 3.1. *Given a finite field \mathbb{F}_q and the polynomials*

($\star\star\star$)
$$x^N - 1 \text{ and } c_1 + c_2x^{a_2} + \dots + c_t x^{a_t},$$

in $\mathbb{F}_q[x]$ with $0 < a_2 < \dots < a_t < N$, $\gcd(N, a_2, \dots, a_t) = 1$, $c_i \neq 0$ for all i , and $N|(q-1)$, there exists a deterministic algorithm with complexity

$$q^{1/4}(\log q)^{O(1)} + 2^{O(t)}(t \log N)^{O(1)} + N^{\frac{t-2}{t-1}+o(1)}(\log q)^{2+o(1)}$$

to decide whether these two polynomials share a root in \mathbb{F}_q . Furthermore, for some $\delta'|N$ with $\delta' \leq N^{\frac{t-2}{t-1}}$ and $\gamma \in \{1, \dots, \delta'\}$, the set of roots of ($\star\star\star$) is equal to the union of a set of cardinality at most $2\gamma N^{\frac{t-2}{t-1}}/\delta'$ and the union of $\delta' - \gamma$ cosets of a subgroup of \mathbb{F}_q^ of order N/δ' .*

Proof: By Lemma 1.11 we can find an integer e such that, if m_2, \dots, m_t are the unique integers in the range $[-\lfloor N/2 \rfloor, \lfloor N/2 \rfloor]$ respectively congruent to ea_2, \dots, ea_t , then $|m_i| < N^{\frac{t-2}{t-1}}$ for each $i \in \{2, \dots, t\}$. This takes $2^{O(t)}(t \log N)^{O(1)}$ deterministic bit operations. By [45], we can then find a generator σ of \mathbb{F}_q^* within $q^{1/4}(\log q)^{O(1)}$ bit operations. For any $\tau \in \mathbb{F}_q^*$, let $\langle \tau \rangle$ denote the multiplicative subgroup of \mathbb{F}_q^* generated by τ .

Now, $x^N - 1$ vanishing is the same as $x \in \langle \sigma^{\frac{q-1}{N}} \rangle$ since $N|(q-1)$. Let $\zeta_N := \sigma^{\frac{q-1}{N}}$ and define $\delta' := \gcd(e, N)$. If $\delta' = 1$ then the map from $\langle \zeta_N \rangle$ to $\langle \zeta_N \rangle$ given by $x \mapsto x^e$ is one-to-one. So finding a solution for ($\star\star\star$) is equivalent to finding $x \in \langle \zeta_N \rangle$ such that $c_1 + c_2x^{ea_2} + \dots + c_t x^{ea_t} = 0$. Thanks to Lemma 1.11, the last equation can be rewritten as the lower degree equation $c_1 + c_2x^{m_2} + \dots + c_t x^{m_t} = 0$, and we may conclude our proof by applying Proposition 2.3.

However, we may have $\delta' > 1$. In which case, the map from $\langle \zeta_N \rangle$ to $\langle \zeta_N \rangle$ given by $x \mapsto x^e$ is no longer one-to-one. Instead, it sends $\langle \zeta_N \rangle$ to a smaller subgroup $\langle \zeta_N^{\delta'} \rangle$ of order N/δ' . We first bound δ' : re-ordering monomials if necessary, we may assume that $m_2 \neq 0$. We then obtain

$$\delta' = \gcd(e, N) \leq \gcd(ea_2, N) = \gcd(m_2, N) \leq |m_2| \leq N^{\frac{t-2}{t-1}}.$$

Any element $x \in \langle \zeta_N \rangle$ can be written as $\zeta_N^i z$ for some $i \in \{0, \dots, \delta' - 1\}$ and $z \in \langle \zeta_N^{\delta'} \rangle$. It is then clear that $x^N - 1 = c_1 + c_2x^{a_2} + \dots + c_t x^{a_t} = 0$ has a root in \mathbb{F}_q^* if and only if there is an $i \in \{0, \dots, \delta' - 1\}$ and a $z \in \langle \zeta_N^{\delta'} \rangle$ with $c_1 + c_2(\zeta_N^i z)^{a_2} + \dots + c_t(\zeta_N^i z)^{a_t} = 0$. Now, $\gcd(e/\delta', N/\delta') = 1$. So the map from $\langle \zeta_N^{\delta'} \rangle$ to $\langle \zeta_N^{\delta'} \rangle$ given by $x \mapsto x^{e/\delta'}$ is one-to-one. By the definition of the m_i , ($\star\star\star$) having a solution is thus equivalent to there being an $i \in \{0, \dots, \delta' - 1\}$ and a $z \in \langle \zeta_N^{\delta'} \rangle$ with $c_1 + c_2\zeta_N^{a_2 i} z^{m_2/\delta'} + \dots + c_t\zeta_N^{a_t i} z^{m_t/\delta'} = 0$. So define the Laurent polynomial

$$f_i(z) := c_1 + c_2(\zeta_N^i)^{a_2} z^{m_2/\delta'} + \dots + c_t(\zeta_N^i)^{a_t} z^{m_t/\delta'}$$

If f_i is identically zero then we have found a whole set of solutions for $(\star\star\star)$: the coset $\zeta_N^i \langle \zeta_N^{\delta'} \rangle$. If f_i is not identically zero then let $\ell := \min_i \min\{m_i/\delta', 0\}$. The polynomial $z^{-\ell} f_i(z)$ then has degree bounded from above by $2N^{\frac{t-2}{t-1}}/\delta'$. Deciding whether the pair of equations $z^{N/\delta'} - 1 = z^{-\ell} f_i(z) = 0$ has a solution for some i takes deterministic time $\delta' \left(N^{\frac{t-2}{t-1}}/\delta'\right)^{1+o(1)} (\log q)^{2+o(1)}$, applying Proposition 2.3 δ' times.

The final statement characterizing the set of solutions to $(\star\star\star)$ then follows immediately upon defining γ to be the number of $i \in \{0, \dots, \delta' - 1\}$ such that f_i is not identically zero. In particular, $\gamma \geq 1$ since $\deg f < N$ and thus f is not identically zero on the order N subgroup of \mathbb{F}_q^* . ■

EXAMPLE 3.2. Consider any polynomial of the form

$$f(x) = c_1 + c_2x + c_3x^{2^{200}+26} + c_4x^{2^{200}+27} \in \mathbb{F}_q[x]$$

where $q := 6(2^{200} + 26) + 1$ (which is a 61-digit prime) and $c_1c_4 \neq 0$. Considering the lattice generated by the vectors $(1, 2^{200} + 26, 2^{200} + 27), (q - 1, 0, 0), (0, q - 1, 0), (0, 0, q - 1)$, it is not hard to see that $(6, 0, 6)$ is a minimal length vector in this lattice. Moreover, $6 \cdot 1 \equiv 6$, $6(2^{200} + 26) \equiv 0$, $6(2^{200} + 27) \equiv 6 \pmod{q - 1}$. Letting σ be any generator of \mathbb{F}_q^* it is clear that any $x \in \mathbb{F}_q^*$ can be written as $x = \sigma^i z$ for some $i \in \{0, \dots, 5\}$ and $z \in \mathbb{F}_q^*$ satisfying $z^{\frac{q-1}{6}} = 1$. So then, we see that solving $f(x) = 0$ is equivalent to finding an $i \in \{0, \dots, 5\}$ and a $z \in \mathbb{F}_q^*$ with

$$\left(c_1 + c_3\sigma^{(2^{200}+26)i}\right) + \left(c_2\sigma^i + c_4\sigma^{(2^{200}+27)i}\right) z^6 = z^{\frac{q-1}{6}} - 1 = 0. \quad \diamond$$

REMARK 3.3. Via fast randomized factoring, we can also pick out a representative from each coset of roots within essentially the same time bound. Note also that it is possible for some of the Laurent polynomials f_i to vanish identically: The polynomial $1 + x - x^2 - x^3$ and the prime $q = 13$, obtained by mimicking Example 3.2, provide one such example (with $\delta' = 6$ and $\gamma = 1$). \diamond

We are now ready to prove our first main theorem.

Proof of Theorem 1.1: Let $\delta := \gcd(q-1, a_2, \dots, a_t)$ and $y = x^\delta$. Then the solvability of f is equivalent to the solvability of the following system of equations:

$$\begin{aligned} c_1 + c_2y^{a_2/\delta} + \dots + c_t y^{a_t/\delta} &= 0 \\ y^{\frac{q-1}{\delta}} &= 1 \end{aligned}$$

Since $\gcd(\frac{a_1}{\delta}, \dots, \frac{a_t}{\delta}, \frac{q-1}{\delta}) = 1$, we can solve this problem via Lemma 3.1 (with $N = \frac{q-1}{\delta}$), within the stated time bound. (Note that $q^{1/4} \leq q^{\frac{t-2}{t-1}}$ for all $t \geq 3$. Also, the computation of $\gcd(q-1, a_2, \dots, a_t)$ is dominated by the other steps of the algorithm underlying Lemma 3.1.) Also, since $y^{\frac{q-1}{\delta}} = 1$, each solution y of the preceding 2×1 system induces exactly δ roots of f in \mathbb{F}_q . So we can indeed efficiently detect roots of f , and the second assertion of Lemma 3.1 gives us the stated characterization of the roots of f . In particular, S_1 is the unique order δ subgroup of \mathbb{F}_q^* , and S_2 is the unique order $\frac{q-1}{\delta'}$ subgroup of \mathbb{F}_q^* (following the notation of the proof of Lemma 3.1).

3.2. The Proof of Corollary 1.3. Deciding whether 0 is a root of all the f_i is trivial, so let us divide all the f_i by a suitable power of x so that all the f_i have a nonzero constant term. Next, concatenate all the nonzero exponents of the f_i into a single vector of length $T \leq k(t-1)$. Applying Lemma 1.11, and repeating our power substitution trick from our proof of Theorem 1.1, we can then reduce to the case where each f_i has degree at most $2q^{1-T^{-1}}$, at the expense of $2^{O(T)}(T \log q)^{O(1)}$ deterministic bit operations.

At this stage, we then simply compute

$$g(x) := ((\dots (\gcd(\gcd(f_1, f_2), f_3), \dots), f_k)$$

via $k - 1$ applications of the Knuth-Schönhage algorithm [10, Ch. 3]. This takes

$$(k - 1) \left(2q^{1-T^{-1}}\right)^{1+o(1)} (\log q)^{1+o(1)}$$

deterministic bit operations. We then conclude via Proposition 2.3, at a cost of $\left(2q^{1-T^{-1}}\right)^{1+o(1)} (\log q)^{2+o(1)}$ bit operations.

Summing the complexities of our steps, we arrive at our stated complexity bound. ■

4. Hardness in One Variable: Proving Theorems 1.5, 1.6, and 1.10.

4.1. The Proof of Theorem 1.5. Thanks to Theorem 1.12 we obtain an immediate ZPP-reduction from 3CNFSAT to the detection of roots in \mathbb{F}_p for systems of univariate polynomials in $\mathbb{F}_p[x]$. By Lemma 2.2 we then obtain a BPP-reduction to 2×1 systems. Let us now describe a ZPP-reduction from 2×1 systems to 1×1 systems.

Suppose $\chi \in \mathbb{F}_q$ is a quadratic non-residue. Clearly, the only root in \mathbb{F}_q^2 of the quadratic form $x^2 - \chi y^2$ is $(0, 0)$. So we can decide the solvability of $f_1(x) = f_2(x) = 0$ over \mathbb{F}_q by deciding the solvability of $f_1^2 - \chi f_2^2$ over \mathbb{F}_q . Finding a usable χ is easily done in ZPP via random-sampling and polynomial-time Jacobi symbol calculation (see, e.g., [4, Cor. 5.7.5 & Thm. 5.9.3, pg. 110 & 113]).

So there is indeed a BPP-reduction from 3CNFSAT to our main problem, and we are done. ■

4.2. The Proof of Theorem 1.6. First note that the hardness of detecting common degree one factors in $\mathbb{F}_p[x]$ (or $\overline{\mathbb{F}}_p[x]$) for *pairs* of polynomials in $\mathbb{F}_p[x]$ follows immediately from Theorem 1.12 and Lemma 2.2: The proof of Theorem 1.5 above already tells us that there is a BPP-reduction from 3CNFSAT to detecting common roots in $\overline{\mathbb{F}}_p$ of pairs of polynomials in $\mathbb{F}_p[x]$. Thanks to Assertion (4) of Theorem 1.12, we also obtain a BPP-reduction to detecting common roots, in \mathbb{F}_p instead, for pairs of polynomials in $\mathbb{F}_p[x]$.

So why does this imply hardness for deciding divisibility by the square of a degree one polynomial in $\overline{\mathbb{F}}_p[x]$ (or $\mathbb{F}_p[x]$)? Assume temporarily that Problem (2) is doable in BPP. Consider then, for any $f, g \in \mathbb{F}_p[x]$, the polynomial $H := (f + ag)(f + bg)$ where $\{a, b\} \subset \mathbb{F}_p$ is a uniformly random subset of cardinality 2. Note that should f and g have a common factor in $\overline{\mathbb{F}}_p[x]$, then H has a repeated factor in $\overline{\mathbb{F}}_p[x]$.

On the other hand, if f and g have no common factor, then $f + ag$ and $f + bg$ clearly have no common factors. Moreover, thanks to Lemma 2.10, the probability that $f + ag$ and $f + bg$ are both square-free — and thus H is square-free — is at least $\left(1 - \frac{2d-1}{q}\right) \left(1 - \frac{2d-2}{q}\right)$.

In other words, to test f and g for common factors, it's enough to check square-freeness of H for random (a, b) .

To conclude, thanks to Theorem 1.12, the pairs of polynomials arising from our BPP-reduction from 3CNFSAT satisfy the hypothesis of Lemma 2.10. Furthermore, thanks to Assertion (1) of Theorem 1.12, our success probability is at least $\left(1 - \frac{2}{11}\right)^2 \geq \frac{2}{3}$, so we are done. ■

4.3. Proving Theorem 1.10. We will need the following proposition, due to Ryan Williams.

PROPOSITION 4.1. [49] Assume, for any Boolean circuit with n inputs and size polynomial in n , that the Circuit Satisfiability Problem can be solved in time $2^{n-\omega(\log n)}$. Then $\text{NEXP} \not\subseteq \text{P/poly}$. ■

We will also need the following lemma, which is implicit in [36]. For completeness, we supply a proof below.

LEMMA 4.2. Given a Boolean circuit with d inputs and L gates, we can find in polynomial time a straight-line program of size $L^{O(1)}$ for a polynomial $f \in \mathbb{F}_2[x]$ such that the circuit is satisfiable if and only if f has a root in \mathbb{F}_2^d .

Proof: A Boolean circuit can be viewed as a straight-line program using Boolean variables and Boolean operations. One can replace Boolean operations by polynomials over \mathbb{F}_2 as follows: $y_1 \wedge y_2 \mapsto y_1 y_2$, $y_1 \vee y_2 \mapsto y_1 + y_2 + y_1 y_2$, and $\neg y_1 \mapsto 1 + y_1$. Hence a straight-line program for a Boolean function of size L with d inputs can be converted into a straight-line program for a polynomial $g(y_0, y_1, \dots, y_{d-1}) \in \mathbb{F}_2[y_0, y_1, \dots, y_{d-1}]$ of size $O(L)$, such that the circuit is satisfiable if and only if the equation $g = 1$ has a root in \mathbb{F}_2^d .

Let $b(x)$ be an irreducible polynomial of degree d over \mathbb{F}_2 . Let α be one root of $b(x)$. Then $\{1, \alpha, \alpha^2, \dots, \alpha^{d-1}\}$ is a basis for \mathbb{F}_2^d over \mathbb{F}_2 . Then any element $x \in \mathbb{F}_2^d$ can be written uniquely as $x = x_0 + x_1 \alpha + \dots + x_{d-1} \alpha^{d-1}$, where $x_i \in \mathbb{F}_2$ for all i . So we obtain the system of linear equations

$$\begin{bmatrix} 1 & \alpha & \dots & \alpha^{d-1} \\ 1 & \alpha^2 & \dots & \alpha^{2(d-1)} \\ 1 & \alpha^4 & \dots & \alpha^{4(d-1)} \\ \vdots & & & \vdots \\ 1 & \alpha^{2^{d-1}} & \dots & \alpha^{2^{d-1}(d-1)} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{d-1} \end{bmatrix} = \begin{bmatrix} x \\ x^2 \\ x^4 \\ \vdots \\ x^{2^{d-1}} \end{bmatrix}.$$

The underlying matrix is Vandermonde and thus non-singular. So we can represent each x_i as a linear combination of $x, x^{2^1}, x^{2^2}, \dots, x^{2^{d-1}}$ over \mathbb{F}_2^d . Note that x^{2^i} has a straight-line program of length $i + 2$. Replacing each x_i by the appropriate linear combination of high powers of x , in the SLP for g , we obtain a SLP for a univariate polynomial h in x , such that $g = 1$ has a root in \mathbb{F}_2^d if and only if $h - 1$ has a root in \mathbb{F}_2^d . Moreover, the straight-line program needs only one constant α that can be constructed in polynomial time. Our lemma follows by setting $f := h - 1$. ■

Proof of Theorem 1.10: From Lemma 4.2, an algorithm as hypothesized in Theorem 1.10 would imply a $2^{\ell-\omega(\log \ell)}$ algorithm for any instance of the Circuit Satisfiability Problem of ℓ inputs and size polynomial in ℓ . By Proposition 4.1, we would then obtain $\text{NEXP} \not\subseteq \text{P/poly}$. ■

Acknowledgements. We would like to thank Igor Shparlinski and Phong Q. Nguyen for insightful comments on an earlier draft of this paper. We also thank the referees for their comments, which helped clarify our paper.

REFERENCES

- [1] SANJEEV ARORA AND BOAZ BARAK, *Computational complexity. A modern approach*. Cambridge University Press, Cambridge, 2009.
- [2] MARTIN AVENDAÑO, ASHRAF IBRAHIM, J. MAURICE ROJAS, AND KORBEN RUSEK, “Faster p -adic Feasibility for Certain Multivariate Sparse Polynomials,” *Journal of Symbolic Computation*, special issue in honor of 60th birthday of Joachim von zur Gathen, vol. 47, no. 4, pp. 454–479 (April 2012).
- [3] MARTIN AVENDAÑO, TERESA KRICK, AND MARTIN SOMBRA, “Factoring bivariate sparse (lacunary) polynomials,” *J. Complexity*, vol. 23 (2007), pp. 193–216.

- [4] ERIC BACH, AND JEFF SHALLIT, *Algorithmic Number Theory, Vol. I: Efficient Algorithms* MIT Press, Cambridge, MA, 1996.
- [5] OSBERT BASTANI, CHRIS HILLAR, DIMITAR POPOV, AND J. MAURICE ROJAS, “*Randomization, Sums of Squares, and Faster Real Root Counting for Tetranomials and Beyond*,” *Randomization, Relaxation, and Complexity in Polynomial Equation Solving*, Contemporary Mathematics, vol. 556, pp. 145–166, AMS Press, 2011.
- [6] SAUGATA BASU, RICHARD POLLACK, AND MARIE-FRANCOISE ROY, *Algorithms in Real Algebraic Geometry*, Algorithms and Computation in Mathematics, vol. 10, Springer-Verlag, 2006.
- [7] ELWYN R. BERLEKAMP, “*Factoring polynomials over large finite fields*,” *Math. Comp.* 24, pp. 713–735 (1970).
- [8] JINGGUO BI, QI CHENG, AND J. MAURICE ROJAS, “*Sub-linear root detection, and new hardness results, for sparse polynomials over finite fields*,” proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC 2013, June 26-29, Boston, USA), pp. 61-68, ACM Press, 2013.
- [9] FREDERIC BIHAN, J. MAURICE ROJAS, AND CASEY STELLA, “*Faster Real Feasibility via Circuit Discriminants*,” proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC 2009, July 28–31, Seoul, Korea), pp. 39–46, ACM Press, 2009.
- [10] PETER BÜRGISSER, MICHAEL CLAUSEN, AND AMIN SHOKROLLAHI, *Algebraic complexity theory*, with the collaboration of Thomas Lickteig, Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], 315, Springer-Verlag, Berlin, 1997.
- [11] RAN CANETTI, JOHN B. FRIEDLANDER, SERGEI KONYAGIN, MICHAEL LARSEN, D. LIEMAN, AND IGOR E. SHPARLINSKI, “*On the statistical properties of Diffie-Hellman distributions*,” *Israel J. Math.* 120 (2000), pp. 23–46.
- [12] DAVID C. CANTOR AND HANS ZASSENHAUS, “*A new algorithm for factoring polynomials over finite fields*,” *Math. Comp.* 36 (1981), no. 154, pp. 587–592.
- [13] J. W. S. CASSLS, *An Introduction to the Geometry of Numbers*, Springer Classics in Mathematics, Springer-Verlag, 1997.
- [14] QI CHENG, SHUHONG GAO, J. MAURICE ROJAS, AND DAQING WAN, “*Sparse Univariate Polynomials with Many Roots over Finite Fields*,” submitted for publication. Also available as Math ArXiv preprint [1411.6346](https://arxiv.org/abs/1411.6346).
- [15] QI CHENG, J. HILL, AND DAQING WAN, “*Counting Value Sets: Algorithm and Complexity*,” ANTS X, Proceedings of the Tenth Algorithmic Number Theory Symposium, The Open Book Series, 1(2013), pp. 235–248.
- [16] J. CHRISTOPHER, “*The Asymptotic Density of Some k -Dimensional Sets*,” *the American Mathematical Monthly*, vol. 63, no. 6 (Jun.–Jul., 1956), pp. 399-401.
- [17] DAVID A. COX, *personal communication*, August, 2004.
- [18] FELIPE CUCKER, PASCAL KOIRAN, AND STEVE SMALE, “*A polynomial time algorithm for Diophantine equations in one variable*,” *J. Symbolic Comput.* 27 (1999), pp. 21–29.
- [19] D. DADUSH, C. PEIKERT, AND SANTOSH VEMPALA, “*Enumerative Lattice Algorithms in any Norm Via M -ellipsoid Coverings*,” *FOCS 2011*, pp. 580–589.
- [20] IOANNIS EMIRIS, AND VICTOR PAN, “*Improved algorithms for computing determinants and resultants*,” *J. Complexity (FOCM 2002 special issue)*, Vol. 21, no. 1, February 2005, pp. 43–71.
- [21] GAREY, MICHAEL R. AND JOHNSON, DAVID S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, A Series of Books in the Mathematical Sciences, W. H. Freeman and Co., San Francisco, Calif., 1979.
- [22] JOACHIM VON ZUR GATHEN, “*Who was who in polynomial factorization*,” *Proceedings of ISSAC 2006* (B. M. Trager, ed.), pp. 2–3, ACM Press, 2006.
- [23] JOACHIM VON ZUR GATHEN, MAREK KARPINSKI, AND IGOR E. SHPARLINSKI, “*Counting curves and their projections*,” *Computational Complexity* 6, no. 1 (1996/1997), pp. 64–99.
- [24] JOACHIM VON ZUR GATHEN AND DANIEL PANARIO, “*Factoring polynomials over finite fields: A survey*,” *J. Symb. Comput.*, 31(1/2):3–17, 2001.
- [25] JOACHIM VON ZUR GATHEN AND VICTOR SHOUP, “*Computing Frobenius maps and factoring polynomials*,” *Computational Complexity* 2:187–224, 1992.
- [26] ISRAEL M. GEL’FAND, MISHA KAPRANOV, AND ANDREI ZELEVINSKY, *Discriminants, Resultants and Multidimensional Determinants*, Birkhäuser, Boston, 1994.
- [27] MARC GIUSTI, AND JOOS HEINTZ, “*La détermination des points isolés et la dimension d’une variété algébrique peut se faire en temps polynomial*,” *Computational Algebraic Geometry and Commutative Algebra (Cortona, 1991)*, Sympos. Math. XXXIV, pp. 216–256, Cambridge University
- [28] RUSSEL IMPAGLIAZZO, VALENTIN KABANETS, AND AVI WIGDERSON, “*In Search of an Easy Witness: Exponential Time vs. Probabilistic Polynomial Time*,” *Journal of Computer and Sys-*

- tem Sciences, 65(4), pp. 672–694, 2002.
- [29] RUSSEL IMPAGLIAZZO, AND R. PATURI, “*The Complexity of k -SAT*,” Journal of Computer and System Sciences, Volume 62, Issue 2, March 2001, pp. 367–375.
- [30] ERICH KALTOFEN, “*Polynomial factorization: a success story*,” In ISSAC 2003 Proc. 2003 Internat. Symp. Symbolic Algebraic Comput. (New York, N.Y., 2003), J. R. Sendra, Ed., ACM Press, pp. 3–4.
- [31] ERICH KALTOFEN AND PASCAL KOIRAN, “*On the complexity of factoring bivariate supersparse (lacunary) polynomials*,” ISSAC05, Proceedings of 2005 International Symposium Symbolic Algebraic Computation, ACM Press, New York, 2005.
- [32] ERICH KALTOFEN AND VICTOR SHOUP, “*Subquadratic-time factoring of polynomials over finite fields*,” Math. Comp. 67 (1998), no. 223, pp. 1179–1197.
- [33] MAREK KARPINSKI AND IGOR E. SHPARLINSKI, “*On the computational hardness of testing square-freeness of sparse polynomials*,” Applied algebra, algebraic algorithms and error-correcting codes (Honolulu, HI, 1999), pp. 492–497, Lecture Notes in Comput. Sci., 1719, Springer, Berlin, 1999.
- [34] KIRAN KEDLAYA, AND CHRISTOPHER UMANS, “*Fast polynomial factorization and modular composition*,” SIAM Journal on Computing, Vol. 40, No. 6, pp. 1767–1802, 2011.
- [35] ALEXANDER KELLEY AND SEAN OWEN, “*Estimating the Number Of Roots of Trinomials over Finite Fields*,” submitted for publication. Also available as Math ArXiv preprint 1510.01758v2.
- [36] A. KIPNIS AND ADI SHAMIR, “*Cryptanalysis of the HFE public key cryptosystem by relinearization*,” Advances in cryptology — CRYPTO ’99 (Santa Barbara, CA), pp. 19–30, Lecture Notes in Comput. Sci. 1666, Springer, Berlin, 1999.
- [37] ARJEN K. LENSTRA, HENDRIK W. LENSTRA (JR.), AND LASZLO LOVÁSZ, “*Factoring polynomials with rational coefficients*,” Math. Ann. 261 (1982), no. 4, pp. 515–534.
- [38] HENDRIK W. LENSTRA, JR., “*Finding Small Degree Factors of Lacunary Polynomials*,” Number Theory in Progress, Vol. 1 (Zakopane-Kóscielisko, 1997), pp. 267–276, de Gruyter, Berlin, 1999.
- [39] ROY MESHULAM, “*An uncertainty inequality for finite abelian groups*,” European J. of Combinatorics, 27 (2006), pp. 63–67.
- [40] PHONG Q. NGUYEN, “*Lattice Reduction Algorithms: Theory and Practice*,” K.G. Paterson (ed.): Eurocrypt 2011, LNCS 6632, pp. 2–6, 2011.
- [41] DAVID ALAN PLAISTED, “*New NP-Hard and NP-Complete Polynomial and Integer Divisibility Problems*,” Theoret. Comput. Sci. 31 (1984), no. 1–2, 125–138.
- [42] Q.I. RAHMAN, AND G. SCHMEISSER, *Analytic Theory of Polynomials*, Clarendon Press, London Mathematical Society Monographs 26, 2002.
- [43] J. MAURICE ROJAS, AND YINYU YE, “*On Solving Sparse Polynomials in Logarithmic Time*,” Journal of Complexity, special issue for the 2002 Foundations of Computation Mathematics (FOCM) meeting, February 2005, pp. 87–110.
- [44] JACOB T. SCHWARTZ, “*Fast Probabilistic Algorithms for Verification of Polynomial Identities*,” J. of the ACM 27, 701–717, 1980.
- [45] IGOR E. SHPARLINSKI, “*On finding primitive roots in finite fields*,” Theoretical Computer Science, Vol. 157, Issue 2, 5 May 1996, pp. 273–275.
- [46] D.E. SMITH, AND M. LATHAM, *The Geometry of René Descartes*, translated from the French and Latin (with a facsimile of Descartes’ 1637 French edition), Dover Publications Inc., New York (1954).
- [47] TERENCE TAO, “*An Uncertainty Principle for Cyclic Groups of Prime Order*,” Math. Res. Lett. 12 (1) (2005), pp. 121–127.
- [48] CHRISTOPHER UMANS, “*Fast polynomial factorization and modular composition in small characteristic*,” STOC’08, pp. 481–490, ACM, New York, 2008.
- [49] RYAN WILLIAMS, “*Improving exhaustive search implies superpolynomial lower bounds*,” in STOC, 231–240, 2010.
- [50] RICHARD ZIPPEL, “*An explicit separation of relativised random polynomial time and relativised deterministic polynomial time*,” Technical report #965, Department of Computer Science, Cornell University, 1989.