J. Symbolic Computation (1990) **10**, 547-570

Constructing normal bases in finite fields

Joachim von zur Gathen
Mark Giesbrecht

Department of Computer Science, University of Toronto
Toronto, Ontario M5S 1A4, Canada
gathen@theory.toronto.edu
mwg@theory.toronto.edu

(*Received* 6 *June* 1989)

An efficient probabilistic algorithm to find a normal basis in a finite field is presented. It can, in fact, find an element of arbitrary prescribed additive order. It is based on a density estimate for normal elements. A similar estimate yields a probabilistic polynomial-time reduction from finding primitive normal elements to finding primitive elements.

1. Introduction

If $\mathbb{F}_q \subseteq \mathbb{F}_{q^n}$ are finite fields, $\alpha \in \mathbb{F}_{q^n}$, and the conjugates $\alpha, \alpha^q, \alpha^{q^2}, \ldots, \alpha^{q^{n-1}}$ of α form a basis for $\mathbb{F}_{q^n}$ as a vector space over $\mathbb{F}_q$, then this is called a *normal basis*. We call α a *normal element* (of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$).

Normal bases are useful for implementing fast arithmetic in $\mathbb{F}_{q^n}$, in particular exponentiation. Of special interest is $q = 2$ and n reasonably large; as an example, the Diffie & Hellman key exchange is based on exponentiation in $\mathbb{F}_{2^n}$. Algorithms and possible MOS implementations are given in Laws & Rushforth 1971, Wang *et al.* 1985, Beth *et al.* 1986, Agnew *et al.* 1988, Stinson 1990.

The basic assumption in that work is that computing qth powers in $\mathbb{F}_{q^n}$ is for free (i.e., of negligible cost compared to a general multiplication in $\mathbb{F}_{q^n}$; only $q = 2$ is considered). The assumption can be justified if a normal element is given, since then for an arbitrary $u = \sum_{0 \le i < n} u_i \alpha^{q^i} \in \mathbb{F}_{q^n}$, with $u_0, \ldots, u_{n-1} \in \mathbb{F}_q$, we have

$$u^q = (\sum u_i \alpha^{q^i})^q = \sum_{0 \le i < n} u_{i-1} \alpha^{q^i}$$

with $u_{-1} = u_{n-1}$. In other words, the coordinates of u^q are a cyclic shift of those of u.

We also consider a natural generalization of normality, that of *additive order* (Ore 1934, see van der Waerden 1966, Lenstra & Schoof 1987). Consider the *Frobenius automorphism*

Part of this work was done while the first author was a Visiting Fellow at the Computer Science Laboratory, Australian National University, Canberra, Australia, and partly supported by the Natural Sciences and Engineering Research Council of Canada, grant A-2514.

$\sigma : \mathbb{F}_{q^n} \to \mathbb{F}_{q^n}$ of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ with $\sigma(\alpha) = \alpha^q$. For $f = \sum f_k x^k \in \mathbb{F}_q[x]$, we consider the $\mathbb{F}_q$-linear map $f(\sigma) = \sum f_k \sigma^k : \mathbb{F}_{q^n} \longrightarrow \mathbb{F}_{q^n}$. The additive order $\mathrm{Ord}(\alpha) \in \mathbb{F}_q[x]$ of α is defined as the monic generator of the principal ideal $\{f \in \mathbb{F}_q[x] : f(\sigma)(\alpha) = 0\}$ in $\mathbb{F}_q[x]$. Since

$$(x^n - 1)(\sigma)(\alpha) = (\sigma^n - id)(\alpha) = \alpha^{q^n} - \alpha = 0$$

for all $\alpha \in \mathbb{F}_q$, $\mathrm{Ord}(\alpha)$ is a divisor of $x^n - 1$. An element $\alpha \in \mathbb{F}_{q^n}$ is normal over $\mathbb{F}_q$ if and only if $\mathrm{Ord}(\alpha) = x^n - 1$.

This paper addresses the question: how can we find a normal element efficiently? More generally, we consider how to find an element of any given additive order. Hensel (1888) pioneered the study of normal bases for finite fields and proved that they always exist. We use his algorithm in Section 2. Eisenstein (1850) had already noted that normal bases always exist. Hensel, and also Ore (1934), determine exactly the number of these bases, and Ore develops the more general concept of additive order. Ore's approach is developed into more constructive proofs of the normal basis theorem in several textbooks (for example, van der Waerden 1966, Section 67, and Albert 1956, Section 4.15); these all use some linear algebra calculations. Schwarz (1988) has given a new proof along these lines, and several recent papers have translated this approach into algorithms. Sidel'nikov (1988) deals with the case where n divides one of p (the characteristic of $\mathbb{F}_q$), $q+1$, or $q-1$ and Stepanov & Shparlinsky (1987) with the case $\gcd(n,q) = 1$ or $q > n$. Semaev (1989) solves the general problem; also he observes that the general case reduces to the case where n is a prime power. These three papers reduce in deterministic polynomial time the construction of a normal basis to factoring polynomials; hence—at the current state of the art for factoring—they yield polynomial-time algorithms (requiring $(n \log q)^{O(1)}$ operations in $\mathbb{F}_q$) that are probabilistic in general, and deterministic if q is small (requiring $(nq)^{O(1)}$ operations in $\mathbb{F}_q$). The best general result to date is Lenstra's (1989) deterministic polynomial-time algorithm, given as a subroutine in his interesting solution of a more difficult problem, namely the construction of explicit isomorphisms between finite fields of the same cardinality[1]. Lenstra's assumptions amount (within deterministic polynomial time) to requiring an irreducible polynomial of degree n as input. His sufficient condition is also necessary since an irreducible polynomial is computed.

We assume that a *description* of $\mathbb{F}_q$ is available; if $q = p^m$ for some prime p and integer m this consists of the binary representation of p, and of the coefficients of some irreducible monic polynomial $w \in \mathbb{F}_p[x]$ of degree m with $\mathbb{F}_q = \mathbb{F}_p[x]/(w)$. This allows us to perform operations in $\mathbb{F}_q$ efficiently and, unless otherwise noted, we count operations in $\mathbb{F}_q$ in the analyses of our algorithms. Lenstra (1989) has a more general notion of "explicit data" for $\mathbb{F}_{q^n}$, allowing an arbitrary "multiplication table" in some basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. One can then recover an irreducible polynomial w as above using linear algebra, or else simulate arithmetic in $\mathbb{F}_{q^n}$ by $O(n^2)$ operations in $\mathbb{F}_q$.

Along with a description of $\mathbb{F}_q$, the input consists of a description of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$, i.e., a monic irreducible polynomial $h \in \mathbb{F}_q[x]$ of degree n, where $\mathbb{F}_{q^n} = \mathbb{F}_q[x]/(h)$. This description then consists of n elements of $\mathbb{F}_q$, and the goal is an algorithm using a number of operations in $\mathbb{F}_q$ which is polynomial in n and $\log q$. We present two algorithms for our problem, in Sections 2 and 4. Both are based on well-known properties of normal bases; Lidl & Niederreiter (1983) is our standard reference.

The first algorithm simply takes a random element in $\mathbb{F}_{q^n}$ and tests whether it is normal

[1] Lenstra (1989), and a first version of this paper appeared independently in May, 1989

over $\mathbb{F}_q$; the test is from Hensel (1888). It works in expected polynomial time if there are sufficiently many normal elements. Section 3 is devoted to showing that indeed these are plentiful; we prove analogues—for polynomials over finite fields—of some standard bounds on number theoretic functions. The density of normal elements in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ is $\Omega(1/\log_q n)$ for $n \geq q^4$. Given a normal element and a divisor $g \in \mathbb{F}_q[x]$ of $x^n - 1$, we note that it is easy to construct an element of $\mathbb{F}_{q^n}$ of additive order g.

In Section 4, we adapt the usual linear algebra approach for finding a normal basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ to the problem of finding an element of prescribed additive order $g \in \mathbb{F}_q[x]$. It requires as additional input the complete factorization of g in $\mathbb{F}_q[x]$. This is a stronger assumption than Lenstra's of having an irreducible polynomial in $\mathbb{F}_q$ of degree n (for $g = x^n - 1$), but the state of the art is identical for both requirements: these data can be furnished in random polynomial time, or in deterministic polynomial time assuming the Extended Riemann Hypothesis (ERH), or deterministically in time $\sqrt{p}\cdot(n \log q)^{O(1)}$, where p is the characteristic of $\mathbb{F}_q$ (see Section 4 for references). We obtain a polynomial-time algorithm for an element of additive order g which is probabilistic for arbitrary values of q and n, and deterministic for "small" $q \leq n$ (in fact, $\operatorname{char} \mathbb{F}_q = n^{O(1)}$ is sufficient), or if one assumes the ERH.

An element $\alpha \in \mathbb{F}_{q^n}$ is *primitive* if and only if every non-zero element of $\mathbb{F}_{q^n}$ is a power of α. No (probabilistic) polynomial-time algorithm is known to test primitivity, or to generate a primitive element. For every n and q, there exists a primitive element of $\mathbb{F}_{q^n}$ which is normal over $\mathbb{F}_q$: a *primitive normal element* (Lenstra & Schoof 1987). Stepanov & Shparlinsky (1989) give a deterministic reduction from finding primitive normal elements to finding primitive elements; their reduction uses time linear in $\log q$, but exponential in n. In Section 6, we give a probabilistic polynomial-time (in $n \log q$) reduction from finding a primitive normal element in $\mathbb{F}_{q^n}$ to finding a primitive elements in $\mathbb{F}_{q^n}$. This is based on further estimates of finite field analogues of number-theoretic functions, given in Section 5.

We think that Lenstra's comment "Although the algorithms presented in this [Lenstra's] paper are not necessarily inefficient, I do not expect that in practice they can compete with the probabilistic algorithms ..." also applies to the methods of Section 4, and expect methods avoiding linear algebra, such as the algorithm of Section 2, to perform better in practice.

At a referee's suggestion, we summarize the results of this paper, as follows:

1. a fast algorithm in Section 2 for computing a normal basis of degree n over $\mathbb{F}_q$, requiring an expected number $O\tilde{\ }(n^2 \log q)$ operations in $\mathbb{F}_q$ with fast arithmetic, and an expected number $O(n^3 \log q)$ operations in $\mathbb{F}_q$ with "naive" arithmetic; this compares favourably with the previously known $O(n^{3.39} \log q)$ and "naive" $O(n^4 \log q)$ operations in $\mathbb{F}_q$ respectively, based on linear algebra;

2. generalizations of both the fast probabilistic algorithm and the slower deterministic algorithm from finding normal elements to finding elements of arbitrary additive order;

3. a (random) polynomial-time reduction of finding primitive normal elements to finding a primitive element;

4. (high) density estimates for normal and primitive normal elements (Sections 3, 5, and 6).

While the algorithms themselves are easily deduced from the literature, our main technical tool, the density estimates, are new.

2. A probabilistic algorithm

We denote by $\nu(n,q)$ the number of normal elements in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. For $g,h \in \mathbb{F}_q[x]$ with $h \neq 0$, we denote by g rem $h \in \mathbb{F}_q[x]$ the remainder of g on division by h: $(g \text{ rem } h) \equiv g \bmod h$ and $\deg(g \text{ rem } h) < \deg h$. The gcd of two polynomials (not both zero) is the *monic* polynomial of largest degree dividing both polynomials.

Let $h \in \mathbb{F}_q[x]$ be monic irreducible of degree n, $\mathbb{F}_{q^n} = \mathbb{F}_q[x]/(h)$, $\alpha = x \bmod h \in \mathbb{F}_{q^n}$, and $(1, \alpha, \alpha^2, \ldots, \alpha^{n-1})$ the standard basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. We want to find a normal basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. In this and the next section, we discuss the following algorithm, based on Hensel's (1888) criterion.

Algorithm A.
Input: A description of $\mathbb{F}_q$, a monic irreducible polynomial $h \in \mathbb{F}_q[x]$ of degree n, an estimate $N \le \nu(n,q)$, and a confidence parameter $\epsilon > 0$.

Output: An element $\alpha \in \mathbb{F}_q[x]/(h) \cong \mathbb{F}_{q^n}$ normal over $\mathbb{F}_q$, or "failure".

1. Set $k = \lceil \log \epsilon / \log(1 - Nq^{-n}) \rceil$, and set a counter c to 1.
2. Choose $\alpha \in \mathbb{F}_{q^n} = \mathbb{F}_q[x]/(h)$ at random.
3. For $1 \le i \le n$, compute $\beta_i = \alpha^{q^i} \in \mathbb{F}_{q^n}$.
4. Set $w = (\sum_{0 \le i < n} \beta_i z^i) \in \mathbb{F}_{q^n}[z]$, and compute $g = \gcd(w, z^n - 1) \in \mathbb{F}_{q^n}[z]$.
5. If $g = 1$, return α and stop. Otherwise increase c by 1. If $c \le k$, go to step 2, else return "failure" and stop.

To estimate the cost, let $M : \mathbb{N} \to \mathbb{R}$ denote a "universal" cost of multiplication, i.e., be such that two polynomials of degree at most n over a ring R can be multiplied in $O(M(n))$ arithmetic operations in R, and two n-bit integers can be multiplied with $O(M(n))$ bit operations. We can choose $M(n) = n \log n \log\log n$ (Schönhage & Strassen 1971, Cantor & Kaltofen 1987). If $g, h \in \mathbb{F}_q[x]$ are polynomials of degree at most n, then g rem h (if $h \neq 0$) can be calculated in $O(M(n))$ operations in $\mathbb{F}_q$, and $\gcd(g,h)$ in $O(M(n)\log n)$ operations (see Aho *et al.* 1974, Section 8.9).

It is convenient to ignore logarithmic factors using the "soft O" notation, introduced by von zur Gathen (1985) and Babai *et al.* (1988):

$$g = O\tilde{}(h) \iff \exists k\ g = O(h(\log h)^k).$$

THEOREM 2.1. *Algorithm A works correctly as described in "Output"; failure occurs with probability at most ϵ. It uses at most k random choices in $\mathbb{F}_{q^n}$, where $k = \lceil \log \epsilon / \log(1 - Nq^{-n}) \rceil$, and $O(M(n)(M(n)\log n + n \log q))$, or $O\tilde{}(n^2 \log q)$, arithmetic operations in $\mathbb{F}_q$ per polynomial tested, for a total of $O\tilde{}(kn^2 \log q)$ operations.*

PROOF. Hensel (1888) proves that α is normal if and only if $g = 1$ in step 4, and hence an eventual output α is correct (see Lidl & Niederreiter 1983, Theorem 2.39).

By assumption $N \leq \nu(n, q)$, and thus the failure probability is at most $(1-Nq^{-n})^k \leq \epsilon$.

We can compute all β_i in steps 3 by repeated squaring in $O(n \log q)$ operations in $\mathbb{F}_{q^n}$. The gcd in step 4 can be calculated in $O(M(n) \log n)$ operations in $\mathbb{F}_{q^n}$. One such operation can be simulated by $O(M(n))$ steps in $\mathbb{F}_q$, for a total of $O(M(n)^2 \log n)$ operations in $\mathbb{F}_q$. □

Recall from the introduction the notion of additive order: if σ is the Frobenius map in $\mathbb{F}_q$ (mapping an element of $\mathbb{F}_{q^n}$ to its qth power), then the additive order $\mathrm{Ord}(\alpha)$ of any $\alpha \in \mathbb{F}_{q^n}$ is the monic polynomial $g \in \mathbb{F}_q[x]$ of minimal degree such that $g(\sigma)(\alpha) = 0$.

PROPOSITION 2.2. *Given a description of $\mathbb{F}_q$, an irreducible polynomial $h \in \mathbb{F}_q[x]$ of degree n, an element $\alpha \in \mathbb{F}_{q^n} = \mathbb{F}_q[x]/(h)$ normal over $\mathbb{F}_q$, and $g \in \mathbb{F}_q[x]$ dividing $x^n - 1$, we can find an element $\beta \in \mathbb{F}_{q^n}$ of additive order g with $O(nM(n) \log q)$ or $O\tilde{~}(n^2 \log q)$ operations in $\mathbb{F}_q$.*

PROOF. Simply compute $\beta = ((x^n - 1)/g)(\sigma)(\alpha) \in \mathbb{F}_{q^n}$. The element β is certainly annihilated by $g(\sigma)$ since $g(\sigma)(\beta) = (x^n - 1)(\sigma)(\alpha) = 0$. To see $\mathrm{Ord}(\beta) = g$, suppose $h(\sigma)(\beta) = 0$ for some $h \in \mathbb{F}_q[x]$. This implies $(h(x^n - 1)/g)(\sigma)(\alpha) = 0$. Since all annihilators of α are divisible by $x^n - 1$, g must divide h. If we compute α^{q^i} for $0 \leq i \leq n - \deg g$ by repeated squaring, then β can be calculated with $O(n \log q)$ operations in $\mathbb{F}_{q^n}$. Each operation in $\mathbb{F}_{q^n}$ can be simulated with $O(M(n))$ operations in $\mathbb{F}_q$ for a total of $O(nM(n) \log q)$ or $O\tilde{~}(n^2 \log q)$ operations in $\mathbb{F}_q$. □

How can we find the required estimate N for $\nu(n, q)$? Corollary 3.6 below gives a general lower bound on $\nu(n, q)$, and Corollary 3.7 the resulting upper bound on the computing time. For an exact calculation, we let $f_1, \ldots, f_r \in \mathbb{F}_q[x]$ be the distinct irreducible factors of $x^n - 1$ in $\mathbb{F}_q[x]$, and $n_i = \deg f_i$. Then

$$\nu(n, q) = q^n (1 - q^{-n_1}) \cdots (1 - q^{-n_r}),$$

since Lidl & Niederreiter (1983), Theorem 3.73, gives $\nu(n, q)/n$ as the number of normal polynomials. We now discuss two ways of calculating $\nu(n, q)$ exactly, an "arithmetic" and a "Boolean" one. Write $n = n'p^e$, with $e \in \mathbb{N}$ and $p = \operatorname{char} \mathbb{F}_q$ not dividing n'. Then $x^n - 1 = (x^{n'} - 1)^{p^e}$, and the (nonzero) derivative $n'x^{n'-1}$ of $x^{n'} - 1$ has no common factors with $x^{n'} - 1$, so that $x^{n'} - 1$ is squarefree, and $x^{n'} - 1 = \prod_{1 \leq i \leq r} f_i$.

Set $h_1 = x^{n'} - 1$. For $i = 1, 2, \ldots, n'$, we compute $g_i = \gcd(x^{q^i} - x, h_{i-1})$ and $h_i = h_{i-1}/g_i$. Then exactly $\deg g_i / i$ many irreducible factors of $x^n - 1$ have degree i (Lidl & Niederreiter 1983, Theorem 3.20), and

$$\nu(n, q) = q^n \cdot \prod_{1 \leq i \leq n'} (1 - q^{-i})^{\deg g_i / i} = q^{n-n'} \prod_{1 \leq i \leq n'} (q^i - 1)^{\deg g_i / i}. \tag{2.1}$$

Of course, h_i is either 1 or irreducible already for $i = \lfloor n'/2 \rfloor + 1$.

PROPOSITION 2.3. *Given $m, n \in \mathbb{N}$, a prime p, and a description of $\mathbb{F}_{q^n} = \mathbb{F}_q = \mathbb{F}_{p^m}$, $\nu(n, q)$ can be calculated in $O(nM(n \log q))$ Boolean operations, plus $O(nM(n) \log q)$ arithmetic operations in $\mathbb{F}_{q^n}$.*

PROOF. The arithmetic cost for the method sketched is $O(nM(n)\log q)$. The Boolean cost is only $O(\log nM(\log n))$ to find e and n', and $O(n'M(n\log q))$ for the integer product in (2.1). $\square$

We now describe a different way to calculate $\nu(n,q)$. Write $n = n'p^e$ as above. For any divisor d of n', let $\tau(d)$ be the order of q modulo d, i.e., the smallest positive integer such that $q^{\tau(d)} \equiv 1 \bmod d$ (with $\tau(1) = 1$). Furthermore, let $\varphi_d \in \mathbb{F}_q[x]$ be the dth cyclotomic polynomial, and $\phi(d)$ the Euler totient function, i.e., the number of integers between 0 and $d-1$ that are relatively prime to d (with $\phi(1) = 1$). Then

$$x^{n'} - 1 = \prod_{d\,|\,n'} \varphi_d,$$

and φ_d has $\phi(d)/\tau(d)$ many distinct monic irreducible factors, each of degree $\tau(d)$ (Lidl & Niederreiter 1983, Theorems 2.45 and 2.47). Since $x^{n'} - 1$ is squarefree, these factors of $x^{n'} - 1$ are all distinct, and

$$\nu(n,q) = q^n \prod_{d\,|\,n'} (1 - q^{-\tau(d)})^{\phi(d)/\tau(d)} = q^{n-n'} \prod_{d\,|\,n'} (q^{\tau(d)} - 1)^{\phi(d)/\tau(d)}, \tag{2.2}$$

where the product is over all divisors d of n' with $1 \le d \le n'$.

To calculate this expression, we proceed as follows.

1. Compute the prime factorization $n' = \pi_1^{\delta_1} \cdots \pi_t^{\delta_t}$ of n', with distinct prime numbers $\pi_1, \ldots, \pi_t$ and positive integers $\delta_1, \ldots, \delta_t$.
2. Compute $q' = q \operatorname{rem} n'$.
3. For each $1 \le i \le t$ and $2 \le j \le \delta_i$, determine first $\tau(\pi_i) = \operatorname{ord}_{\pi_i}(q')$ and $\phi(\pi_i) = \pi_i - 1$, and then $\tau(\pi_i^j) = \operatorname{ord}_{\pi_i^j}(q')$ and $\phi(\pi_i^j) = (\pi_i - 1)\pi_i^{j-1}$. [Note that $\gcd(q, \pi_i) = 1$, and $\tau(\pi_i^j)$ is either $\tau(\pi_i^{j-1})$ or $\pi_i\tau(\pi_i^{j-1})$ for all i, j.]
4. For each divisor $d = \pi_1^{\epsilon_1} \cdots \pi_t^{\epsilon_1}$ of n', with $0 \le \epsilon_j \le \delta_j$ for all j, compute $\phi(d) = \phi(\pi_1^{\epsilon_1}) \cdots \phi(\pi_t^{\epsilon_t})$ and $\tau(d) = \operatorname{lcm}(\tau(\pi_1^{\epsilon_1}), \ldots, \tau(\pi_t^{\epsilon_t}))$.
5. Compute $\nu(n,q)$ from (2.2).

PROPOSITION 2.4. *Given $m, n \in \mathbb{N}$, a prime p, $q = p^m$, and $\epsilon > 0$, $\nu(n,q)$ can be calculated in $O(n^\epsilon M(n\log q))$ Boolean operations.*

PROOF. We use a trivial method with at most $\sqrt{a}$ trial divisions to factor an integer a, and factor each $\pi_i - 1$ in step 3 to calculate $\tau(\pi_i)$. Step 1 and these factorizations can be done with $O(\sqrt{n}M(\log n) \cdot t)$ bit operations, and step 2 with $O(M(\log(nq)))$ operations. The remaining calculations in steps 3 and 4 only require arithmetic operations on integers with $O(\log n)$ bits, and thus time $(\log n)^{O(1)}$. The number of divisors of n' is $O(n^\epsilon)$ for any $\epsilon > 0$ (Hardy & Wright 1962, Theorem 317). The dominating cost is in computing the products and powers of step 5. These can be calculated, via repeated squaring, with

$$O\left(\log(n - n') + \sum_{d|n'} \left(\log \tau(d) + \log \frac{\phi(d)}{\tau(d)}\right)\right),$$

or $O(n^\epsilon)$ operations on integers with at most $n \log q$ bits. $\square$

This result avoids the arithmetic cost of Proposition 2.2. Note that we work in a different model now, not using arithmetic in $\mathbb{F}_q$ at all, but only Boolean operations with a binary representation of q and n as input.

3. The probability of being normal

Let q be a prime power and $n > 1$. We will show that a randomly chosen element of $\mathbb{F}_{q^n}$ is normal over $\mathbb{F}_q$ with probability $\Omega(1/\log_e n)$.

We denote by I_q the set of monic irreducible polynomials of positive degree in $\mathbb{F}_q[x]$. For $m \geq 1$, let $N_q(m)$ be the number of monic irreducible polynomials of degree m in $\mathbb{F}_q[x]$, and

$$\frac{q^m}{m} - \frac{q(q^{m/2}-1)}{m(q-1)} \leq N_q(m) \leq \frac{q^m}{m} \tag{3.1}$$

(Lidl & Niederreiter 1983, Exercises 3.26 and 3.27). For $f \in \mathbb{F}_q[x]$ and $f \notin \mathbb{F}_q$, let

$$|f| = \#(\mathbb{F}_q[x]/(f)) = q^{\deg f}$$

be the number of elements of the residue class ring of $\mathbb{F}_q[x]$ modulo f. The analogue of Euler's totient function for integers is the number $\Phi_q(f)$ of polynomials in $\mathbb{F}_q[x]$ of smaller degree than $f \in \mathbb{F}_q[x]$ which are relatively prime to f. This is the number of units in $\mathbb{F}_q[x]/(f)$, and (Lidl & Niederreiter 1983, Lemma 3.69 and Theorem 3.73)

$$\begin{aligned} \Phi_q(f) &= |f| \prod_{\substack{g \in I_q \\ g \mid f}} (1 - |g|^{-1}), \\ \nu(n,q) &= \Phi_q(x^n - 1). \end{aligned} \tag{3.2}$$

Our objective is to give a lower bound of $\Omega(1/\log_e n)$ on $\Phi_q(f)/|f|$ for an $f \in \mathbb{F}_q[x]$ of degree n. This immediately shows the required lower bound for normal elements.

We will make use of the following lemma adapted from Apostol (1976), Theorem 3.2(a).

LEMMA 3.1. *For $x \geq 1$*

$$\sum_{n \leq x} \frac{1}{n} \leq \log_e x + C + 1/x$$

where $C \approx 0.577216$ is Euler's constant.

PROOF. Using Euler's summation formula (see Apostol 1976, Theorem 3.1), we find

$$\begin{aligned} \sum_{n \leq x} \frac{1}{n} &= \int_1^x \frac{dt}{t} - \int_1^x \frac{t - \lfloor t \rfloor}{t^2} dt + 1 - \frac{x - \lfloor x \rfloor}{x} \\ &\leq \log_e x - \int_1^x \frac{t - \lfloor t \rfloor}{t^2} dt + 1 = \log_e x + 1 - \int_1^\infty \frac{t - \lfloor t \rfloor}{t^2} dt + \int_x^\infty \frac{t - \lfloor t \rfloor}{t^2} dt \\ &\leq \log_e x + 1 - \int_1^\infty \frac{t - \lfloor t \rfloor}{t^2} dt + \int_x^\infty \frac{1}{t^2} dt = \log_e x + 1 - \int_1^\infty \frac{t - \lfloor t \rfloor}{t^2} dt + \frac{1}{x} \\ &= \log_e x + C + \frac{1}{x}, \end{aligned}$$

where

$$C = 1 - \int_1^\infty \frac{t - \lfloor t \rfloor}{t^2} dt = \lim_{x \to \infty} \left(\sum_{n \leq x} \frac{1}{n} - \log_e x \right)$$

is Euler's constant (see Apostol 1976, pp. 53). □

We will need the following two lemmas.

LEMMA 3.2. *If*

$$V(q)=\sum_{i\geq 2}\sum_{d\geq 1}\frac{1}{idq^{d(i-1)}},$$

then for any $q\geq 2$ we have $V(q)\leq q^{-1}$.

PROOF. By expanding the sum we can write $V(q)=U(q)+E(q)$ where

$$U(q)=\frac{1}{2q}+\frac{7}{12q^2}+\frac{5}{12q^3}+\frac{59}{120q^4}+\frac{4}{15q^5}+\frac{233}{504q^6}+\frac{11}{56q^7}+\frac{11}{60q^8}+\frac{43}{180q^9},$$

$$E(q)=\sum_{\substack{d,i\in\mathbb{N}\\ di\geq 10}}\frac{1}{(i+1)dq^{id}}<\sum_{\substack{d,i\in\mathbb{N}\\ di\geq 10}}\frac{1}{idq^{id}}.$$

For $n\geq 10$ there exist less than n pairs $(i,d)\in\mathbb{N}^2$ such that $i\cdot d=n$. It follows that

$$E(q)<\sum_{n\geq 10}\frac{1}{q^n}=\frac{1}{q^9(q-1)},$$

and that $V(q)\leq q^{-1}W(q)$, where $W(q)=qU(q)+1/(q^8(q-1))$. Since W is a decreasing function of q and $W(2)=321833/322560<1$, $V(q)\leq 1/q$ for all $q\geq 2$. □

LEMMA 3.3. *Let*

$$W(x)=\prod_{\substack{|g|\leq x\\ g\in I_q}}(1-|g|^{-1}).$$

For any $x\geq q$, we have

$$W(x)>\frac{c-c/q}{\log_q x}-\frac{c}{\log_q^2 x}$$

where $c=e^{-C}\approx 0.56146$ and C is Euler's constant.

PROOF. Let $b=\log_q x$. Since $x\geq q$, $W(x)\neq 0$ and we can consider the logarithm of $W(x)$:

$$\begin{aligned}\log_e W(x) &= \sum_{\substack{|g|\leq x\\ g\in I_q}}\log_e(1-|g|^{-1})=\sum_{1\leq d\leq b}N_q(d)\log_e(1-q^{-d})\\ &\geq \sum_{1\leq d\leq b}\frac{q^d\log_e(1-q^{-d})}{d}=-\sum_{1\leq d\leq b}\sum_{i\geq 1}\frac{q^d}{diq^{di}}\\ &= -\sum_{i\geq 1}\sum_{1\leq d\leq b}\frac{1}{diq^{d(i-1)}}>-\sum_{1\leq d\leq b}\frac{1}{d}-\sum_{i\geq 2}\sum_{d\geq 1}\frac{1}{diq^{d(i-1)}}\\ &\geq -\log_e b-C-b^{-1}-q^{-1}\end{aligned}$$

using Lemmas 3.1 and 3.2 and $\log_e(1-t)=-\sum_{i\geq 1}t^i/i$ for any $t<1$. This implies $W(x)>e^{-C}\exp(-q^{-1}-b^{-1})/b$, and using the fact that $\exp(u)\geq 1+u$ for all $u\in\mathbb{R}$, we see

$$W(x)>\frac{c}{b}(1-q^{-1}-b^{-1})=\frac{c-c/q}{\log_q x}-\frac{c}{\log_q^2 x}.\quad\square$$

THEOREM 3.4. *Let q be a prime power, and let $f \in \mathbb{F}_q[x]$ have degree $n \geq 1$. Then $\Phi_q(f)/|f| \leq 1 - q^{-n}$. If $1 \leq n \leq q^4$ then $\Phi_q(f)/|f| > 1/34$, and if $n > q^4$, then*

$$\frac{\Phi_q(f)}{|f|} > \frac{c}{\log_q n} \cdot \left(1 - q^{-1} - \frac{2 - q^{-1}}{\log_q n}\right) > \frac{1}{16 \log_q n},$$

where $c = e^{-C}$ and C is Euler's constant.

PROOF. For the upper bound,

$$\frac{\Phi_q(f)}{|f|} = \prod_{\substack{g|f \\ g \in I_q}} (1 - |g|^{-1}) \leq 1 - |f|^{-1},$$

and equality is achieved when f is irreducible in $\mathbb{F}_q[x]$. To show the lower bound, for $1 \leq n \leq q$,

$$\begin{aligned} \frac{\Phi_q(f)}{|f|} &= \prod_{\substack{g|f \\ g \in I_q}} (1 - |g|^{-1}) \geq \prod_{\substack{g|f \\ g \in I_q}} (1 - q^{-1}) \\ &\geq (1 - q^{-1})^n \geq (1 - q^{-1})^q \geq \frac{1}{4}, \end{aligned}$$

since $(1 - u^{-1})^u$ is an increasing function of u for $u > 1$, and $q \geq 2$. If $n > q$, we write

$$\frac{\Phi_q(f)}{|f|} = \prod_{\substack{g|f \\ g \in I_q}} (1 - |g|^{-1}) = P_1 \cdot P_2,$$

where

$$P_1 = \prod_{\substack{g|f \\ g \in I_q \\ |g| \leq n}} (1 - |g|^{-1}), \qquad P_2 = \prod_{\substack{g|f \\ g \in I_q \\ |g| > n}} (1 - |g|^{-1}).$$

We begin by examining P_2, and write

$$P_2 > \prod_{\substack{g|f \\ g \in I_q \\ |g| > n}} (1 - n^{-1}) = (1 - n^{-1})^\gamma,$$

where $\gamma = \#\{g \in I_q : g|f, |g| > n\}$. Since

$$q^n = |f| \geq \prod_{\substack{g|f \\ g \in I_q}} |g| \geq \prod_{\substack{g|f \\ g \in I_q \\ |g| > n}} |g| > n^\gamma,$$

we find $\gamma \leq n/\log_q n$ and

$$P_2 > \left(1 - \frac{1}{n}\right)^{n/\log_q n}$$

which is an increasing function of n for fixed q. We divide our analysis of the case $n > q$ into two subcases: $q < n \leq q^4$ and $q^4 < n$. When $n > q$ then $P_2 > (1 - q^{-1})^q \geq 1/4$. For a better estimate when $n > q^4$, we use the fact that

$$\left(1 - \frac{1}{n}\right)^{\frac{n}{\log_q n}} > 1 - \frac{1}{\log_q n},$$

which is obtained by raising each side to the power of $\log_q n$. Thus, for $n > q^4$,

$$P_2 > 1 - \frac{1}{\log_q n}.$$

We now bound the function P_1 for $n > q$ by observing

$$P_1 = \prod_{\substack{g \in I_q \\ g|f \\ |g| \le n}} (1 - |g|^{-1}) > \prod_{\substack{g \in I_q \\ |g| \le n}} (1 - |g|^{-1}).$$

For $q < n \le q^4$, this gives us

$$P_1 \ge \prod_{1 \le i \le 4} \left(1 - \frac{1}{q^i}\right)^{N_q(i)} = \left(1 - \frac{1}{q}\right)^q \left(1 - \frac{1}{q^2}\right)^{\frac{q^2-q}{2}} \left(1 - \frac{1}{q^3}\right)^{\frac{q^3-q}{3}} \left(1 - \frac{1}{q^4}\right)^{\frac{q^4-q^2}{4}},$$

which is an increasing function of q, giving $P_1 > 0.118$ since $q \ge 2$. Thus, for $q < n \le q^4$, $\Phi_q(f)/|f| > 1/34$. For $n > q^4$,

$$P_1 > \frac{c - c/q}{\log_q n} - \frac{c}{\log_q^2 n}$$

by Lemma 3.3, so in this case,

$$\begin{aligned} \frac{\Phi_q(f)}{|f|} &= P_1 \cdot P_2 > \left(\frac{c - c/q}{\log_q n} - \frac{c}{\log_q^2 n}\right)\left(1 - \frac{1}{\log_q n}\right) \\ &= \frac{q-1}{q} \cdot \frac{c}{\log_q n} - \frac{2c - c/q}{\log_q^2 n} + \frac{c}{\log_q^3 n} > \frac{q-1}{q} \cdot \frac{c}{\log_q n} - \frac{2c - c/q}{\log_q^2 n}. \end{aligned}$$

This proves the first inequality claimed for $n > q^4$. Since $c = e^{-C} > 1/2$, and $\log_q n > 4$ when $n > q^4$, we have

$$\begin{aligned} \frac{\Phi_q(f)}{|f|} &> \frac{q-1}{q} \cdot \frac{c}{\log_q n}\left(1 - \frac{2q-1}{(q-1)\log_q n}\right) \\ &> \frac{q-1}{q} \cdot \frac{c}{\log_q n}\left(1 - \frac{2q-1}{4(q-1)}\right) > \frac{1}{16 \log_q n}. \quad \square \end{aligned}$$

A similar lower bound also holds for the integer Euler function ϕ, which we will make use of in Section 6.

FACT 3.5. *For $n \ge 3$,*

$$\frac{\phi(n)}{n} > \frac{c}{\log_e \log_e n} \cdot \left(1 - \frac{1.41}{\log_e^2 \log_e n}\right),$$

where $c = e^{-C} \approx 0.56146$.

PROOF. From Rosser & Schoenfeld (1962), (3.41) and (3.42), we have

$$\frac{\phi(n)}{n} > \frac{1}{\lambda/c + c_1/\lambda} = \frac{c}{\lambda} \cdot \left(1 - \frac{c_1 c}{\lambda^2 + c_1 c}\right) > \frac{c}{\lambda}\left(1 - \frac{c_1 c}{\lambda^2}\right),$$

where $\lambda = \log_e \log_e n$ and $c_1 = 2.5036$. $\square$

This brings us to our bound on the probability of being normal.

Corollary 3.6. *For q a prime power and $n > 1$, the probability $\kappa = \nu(q,n)q^{-n}$ of an element chosen randomly from $\mathbb{F}_{q^n}$ being normal over $\mathbb{F}_q$ satisfies $\kappa \le 1 - q^{-1}$. If $n \le q^4$, then $\kappa \ge 1/34$ and, if $n \ge q^4$, then*

$$\kappa > \frac{c}{\log_q n} \cdot \left(1 - q^{-1} - \frac{2 - q^{-1}}{\log_q n}\right) > \frac{1}{16 \log_q n},$$

where $c = e^{-C}$ and C is Euler's constant.

Proof. Applying Theorem 3.4 with $f = x^n - 1$, and using the fact that $\kappa = \Phi_q(x^n-1)/q^n$ (Lidl & Niederreiter 1983, Theorem 3.73), the lower bound follows immediately. We see from (3.2) that $\Phi_q(x^n - 1)/q^n$ is maximized when $x^n - 1$ has only one irreducible factor; i.e., it is a power of $x - 1$ or, equivalently, n is a power of $\operatorname{char} \mathbb{F}_q$. Thus we have

$$\Phi_q(x^n - 1)/q^n \le \Phi_q((x-1)^n) = 1 - q^{-1},$$

and equality is achieved when n is a power of $\operatorname{char} \mathbb{F}_q$. □

Corollary 3.7. *Let $n, q \in \mathbb{N}$, q a prime power, $\rho = 1/34$ if $n \le q^4$, and $\rho = 1/(16 \log_q n)$ if $n > q^4$. Then*

(i) We can choose $N = \rho q^n$ in Algorithm A. Then Theorem 2.1 holds with $k \le 1 + 2\rho^{-1} \log_e \epsilon^{-1}$, which is $O(\log_e \epsilon^{-1} \log_q n)$ for $n \ge q^4$.

(ii) Given a description of $\mathbb{F}_q$ and an irreducible polynomial of degree n in $\mathbb{F}_q[x]$, a normal basis for $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ can be constructed by a probabilistic (Las Vegas) algorithm with failure probability at most ϵ, using $O\tilde{~}(n \log_e \epsilon^{-1})$ random choices in $\mathbb{F}_q$, and $O\tilde{~}(n^2 \log q \log_e \epsilon^{-1})$ arithmetic operations in $\mathbb{F}_q$, if $\epsilon \le q^{-1}$.

(iii) Given only a description of $\mathbb{F}_q$ and an integer $n \ge 2$, we can probabilistically construct a normal basis for $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ using an expected number $O\tilde{~}(n^2 \log q)$ operations in $\mathbb{F}_q$.

Proof. We use the notation of Section 2. By Corollary 3.6, we have $\nu(n,q) \ge \rho q^n$. With $N = \rho q^n$, we have in Theorem 2.1

$$\frac{k-1}{\log_e \epsilon^{-1}} \le \frac{1}{\log_e \left(\frac{1}{1-\rho}\right)} \le \frac{1}{\log_e(1+\rho)} \le \frac{1}{\rho - \frac{\rho^2}{2}} = \frac{1}{\rho} \cdot \frac{2}{2-\rho} < \frac{2}{\rho},$$

since $\rho < 1$. This shows parts (i) and (ii). The algorithm of Rabin (1980) finds an irreducible polynomial of degree n in an expected number $O\tilde{~}(n^2 \log q)$ operations in $\mathbb{F}_q$. We then use such an irreducible polynomial with Algorithm A and $\epsilon = q^{-1}$. This will find a normal element $\alpha \in \mathbb{F}_{q^n}$ over $\mathbb{F}_q$ in an expected number $O(n^2 \log q)$ operations in $\mathbb{F}_q$. Then $\alpha, \alpha^q, \ldots, \alpha^{q^{n-1}}$, forms the desired normal basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$, and this basis can be computed from α using repeated squaring, with $O\tilde{~}(n^2 \log q)$ operations in $\mathbb{F}_q$. □

Applying Proposition 2.2 immediately yields the following.

Corollary 3.8. *Given a description of $\mathbb{F}_q$, a monic irreducible polynomial $h \in \mathbb{F}_q[x]$ of degree n and a divisor $g \in \mathbb{F}_q[x]$ of $x^n - 1$, we can probabilistically find an element of additive order g in $\mathbb{F}_{q^n}$ with an expected number $O\tilde{~}(n^2 \log q)$ operations in $\mathbb{F}_q$.*

4. A deterministic construction

In this section, we implement a different polynomial-time algorithm for finding a normal basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. As in the algorithm of the previous section, we assume that we have a description of $\mathbb{F}_q$ and a monic irreducible polynomial $h \in \mathbb{F}_q[x]$ of degree n, and set $\mathbb{F}_{q^n} = \mathbb{F}_q[x]/(h)$. Furthermore, we assume we have a complete factorization of $x^n - 1$. We want to find a normal basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. When q is small, say $q \le n$, then this algorithm has an efficient deterministic variant. The following argument is influenced by the proof of Theorem 2.35 in Lidl & Niederreiter (1983).

Recall the definition of additive order from the introduction. As in Section 2, we write $x^n - 1 = f_1^{p^e} \cdots f_r^{p^e}$ with $f_1, \ldots, f_r \in \mathbb{F}_q[x]$ irreducible monic and pairwise distinct, $e \in \mathbb{N}$ and p^e the largest power of $p = \operatorname{char} \mathbb{F}_q$ dividing n. For $1 \le i \le r$ and $0 \le j \le p^e$, let $V_{i,j} = \ker f_i^j(\sigma)$ be the nullspace of the linear mapping $f_i^j(\sigma)$. We will use the following lemma.

Lemma 4.1. *In the above notation, we have for* $0 \le i, k \le r$ *and* $0 \le j, l, \le p^e$

(i) $V_{i,j} \subseteq V_{i,j+1}$,

(ii) $\dim V_{i,j} = j \deg f_i$,

(iii) if $i \ne k$, *then* $V_{i,j} \cap V_{k,l} = \{0\}$,

(iv) $\mathbb{F}_{q^n} = \bigoplus_{1 \le i \le r} V_{i,p^e}$.

Proof. (i) is trivial. Since $\gcd(f_i, f_k) = 1$ in (iii), with the Euclidean Algorithm we can find $s, t \in \mathbb{F}_q[x]$ such that $sf_i^j + tf_k^l = 1$. If $\alpha \in V_{i,j} \cap V_{k,l}$, then

$$0 = sf_i^j(\sigma)(\alpha) + tf_k^h(\sigma)(\alpha) = (sf_i^j + tf_k^h)(\sigma)(\alpha) = id(\alpha) = \alpha.$$

This shows (iii). Since $f_i(\sigma)$ is a polynomial of degree $q^{\deg f_i}$, inducing an $\mathbb{F}_q$-linear map, and

$$V_{i,j} = \{\alpha \in \mathbb{F}_{q^n} : f_i(\alpha) \in V_{i,j-1}\},$$

we have inductively for $1 \le j \le p^e$

$$\dim V_{i,j} \le \deg f_i + \dim V_{i,j-1} \le j \deg f_i. \tag{4.1}$$

Again from the Euclidean algorithm, there exist $s_1, \ldots, s_r \in \mathbb{F}_q[x]$ such that

$$1 = \sum_{1 \le i \le r} s_i \cdot (x^n - 1)/f_i^{p^e}.$$

Then for any $\alpha \in \mathbb{F}_{q^n}$,

$$\alpha = \sum_{1 \le i \le r} \Big((s_i \cdot (x^n - 1)/f_i^{p^e})(\sigma)(\alpha) \Big),$$

and $(s_i \cdot (x^n - 1)/f_i^{p^e})(\sigma)(\alpha) \in V_{i,p^e}$ for all i. Thus $\mathbb{F}_{q^n} \subseteq \bigoplus_{1 \le i \le r} V_{i,p^e}$. From

$$n = \dim \ker(x^n - 1)(\sigma) \le \sum_{1 \le i \le r} \dim V_{i,p^e} \le \sum_{1 \le i \le r} p^e \deg f_i = n, \tag{4.2}$$

we conclude that equality holds everywhere in (4.2) and (4.1). (ii) and (iv) follow. $\square$

We now describe an algorithm that finds an element of prescribed order. As a special case, we can use this to find normal elements. Suppose that $g \in \mathbb{F}_q[x]$ is a factor of $x^n - 1$, and $g = \prod_{1 \le i \le r} f_i^{e_i}$ a complete factorization, with $f_1, \ldots, f_r \in \mathbb{F}_q[x]$ as above, and $e_1, \ldots, e_r \in \mathbb{N}$. We may assume that $e_1, \ldots, e_s \ge 1$ and $e_{s+1} = \cdots = e_r = 0$, for some $s \le r$. Let $V = \ker(g(\sigma)) \subseteq \mathbb{F}_q^n$ be the nullspace of the linear mapping $g(\sigma)$, and for $1 \le i \le s$ and $e_i \ge 1$, let

$$V_i' = V_{i,e_i - 1} \subseteq V_i = V_{i,e_i} \subseteq V.$$

Thus $f_i^{e_i}(\sigma)(\alpha) = 0$ for all $\alpha \in V_i$, and Lemma 4.1 (iv) implies that

$$V = \bigoplus_{1 \le i \le s} V_i.$$

By (i) and (ii), V_i' is a proper subspace of V_i, and by (iii), $f_j^{e_j}(\sigma)(\alpha) \ne 0$ for all $\alpha \in V_i \setminus \{0\}$ with $j \ne i$. Now choose some $\alpha_i \in V_i \setminus V_i'$ for each $i \le s$, and set $\beta = \sum_{1 \le i \le s} \alpha_i$. Thus $\mathrm{Ord}(\alpha_i) = f_i^{e_i}$ and $(g/f_i)(\sigma)(\alpha_i) \ne 0$. Then for any $j \le s$ we have

$$\begin{aligned} \frac{g}{f_j}(\sigma)(\beta) &= \sum_{1 \le i \le s} \frac{g}{f_j}(\sigma)(\alpha_i) \\ &= \sum_{i \ne j} \frac{g}{f_j \cdot f_i^{e_i}} \cdot f_i^{e_i}(\sigma)(\alpha_i) + \frac{g}{f_j}(\sigma)(\alpha_j) \\ &= \frac{g}{f_j}(\sigma)(\alpha_j) \ne 0. \end{aligned}$$

Since $\beta \in \bigoplus_{1 \le i \le s} V_i = V$, we have $\mathrm{Ord}(\beta) = g$. So we have the following algorithm.

Algorithm B.

Input: A description of $\mathbb{F}_q$, a monic irreducible polynomial $h \in \mathbb{F}_q[x]$ of degree n, a factor $g \in \mathbb{F}_q[x]$ of $x^n - 1$, and and a complete factorization $g = \prod_{1 \le i \le s} f_i^{e_i}$ of g, with $e_1, \ldots, e_s \ge 1$.

Output: An element $\beta \in \mathbb{F}_{q^n} = \mathbb{F}_q[x]/(h)$ with $\mathrm{Ord}(\beta) = g$.

1. Let $\alpha = x \bmod h \in \mathbb{F}_{q^n}$ be the standard generator of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. Compute the entries of the matrix $\sigma = (s_{ij})_{0 \le i,j < n} \in \mathbb{F}_q^{n \times n}$ in the standard basis $(1, \alpha, \alpha^2, \ldots, \alpha^{n-1})$, given by $\alpha^{iq} = \sum_{0 \le j < n} s_{ij} \alpha^j$ for $0 \le i < n$.
2. For $1 \le i \le s$, compute a basis $(v_{i,1}, \ldots, v_{i,k_i})$ of $V_i' = \ker f_i^{e_i - 1}(\sigma)$, and a basis $(v_{i,1}, \ldots, v_{i,k_i}, v_{i,k_i+1}, \ldots, v_{i,l_i})$ of $V_i = \ker f_i^{e_i}(\sigma)$. [We have $k_i = (e_i - 1)\deg f_i$, and $l_i = e_i \deg f_i$.]
3. Set $b = \sum_{1 \le j \le r} v_{j,k_i+1} = (b_0, \ldots, b_{n-1}) \in \mathbb{F}_q^n$.
4. Return $\beta = \sum_{0 \le i < n} b_i \alpha^i$.

Let $MM = MM_{\mathbb{F}_q} : \mathbb{N} \to \mathbb{R}$ be the cost of *matrix multiplication*, i.e., such that the product of two $n \times n$-matrices over $\mathbb{F}_q$ can be computed with $O(MM(n))$ operations in $\mathbb{F}_q$. We can choose $MM(n) = n^{2.376}$ (Coppersmith & Winograd 1990). Systems of n linear equations in n unknowns can be solved with $O(MM(n))$ operations.

Theorem 4.2. *Given the input for Algorithm B, it returns an element $\beta \in \mathbb{F}_{q^n}$ with* $\mathrm{Ord}(\beta) = g$. *It can be performed with $O(nM(n)\log q + nMM(n))$ operations in $\mathbb{F}_q$.*

Proof. Correctness of the algorithm has been shown. For the timing estimate, we note that the coordinates of all α^{iq} can be computed in $O(n \log q)$ arithmetic operations in $\mathbb{F}_{q^n}$. One operation in $\mathbb{F}_{q^n}$ can be implemented with $O(M(n))$ operations in $\mathbb{F}_q$.

The matrices $f_i(\sigma)$, $f_i^{e_i-1}(\sigma)$ and $f_i^{e_i}(\sigma)$ can be calculated with at most $n_i+2\lfloor \log e_i \rfloor+1$ matrix multiplications. Since $\sum_{1\le i\le r}(n_i + \log e_i) \le n' + n' \log e_i \le n$, the cost for these matrix products is at most $2nMM(n)$. The required bases for nullspaces can be found in $O(nMM(n))$ operations. □

In order to apply the theorem for the construction of normal bases (i.e., $g = x^n - 1$), given only a description of $\mathbb{F}_q$ and n, we have to find an irreducible polynomial of degree n, and factor $x^n - 1$. We make use of the following results.

An irreducible polynomial in $\mathbb{F}_q[x]$ of degree n can be computed with

(a) an expected number $O\tilde{}(n^2 \log q)$ of operations in $\mathbb{F}_q$ by a probabilistic algorithm (Rabin 1980),

(b) $(n \log q)^{O(1)}$ operations deterministically, assuming the ERH (Adleman & Lenstra 1986),

(c) $O\tilde{}(n^4\sqrt{q})$ deterministically (Shoup 1990b).

The irreducible factors of $x^{n'} - 1$ can be found with

(A) an expected number $O\tilde{}(n^2 \log q)$ of operations in $\mathbb{F}_q$ by a probabilistic algorithm (Ben-Or 1981, Cantor & Zassenhaus 1981),

(B) $(n \log q)^{O(1)}$ operations by a deterministic algorithm, assuming the ERH (Huang 1985),

(C) $O\tilde{}(n^2\sqrt{q})$ operations by a deterministic algorithm (Shoup 1990a).

Corollary 4.3. *Given $n \in \mathbb{N}$, $g \in \mathbb{F}_q[x]$ dividing $x^n - 1$, and a description of $\mathbb{F}_q$, one can compute an element of $\mathbb{F}_{q^n}$ with additive order g, (and with $g = x^n - 1$, a normal basis of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$) with*

(i) an expected $O\tilde{}(n^2 \log q + nMM(n))$ operations in $\mathbb{F}_q$ by a probabilistic algorithm,

(ii) $(n \log q)^{O(1)}$ operations deterministically, assuming the ERH,

(iii) $O\tilde{}(n^4\sqrt{q})$ deterministically.

In fact, (iii) also holds with $O\tilde{}((mn)^4\sqrt{p})$ operations in $\mathbb{F}_p$ if $q = p^m$.

Corollary 3.7(iii) compares favourably to (i) above, as does Corollary 3.8 to Theorem 4.2 for computing elements of given additive order.

As pointed out in the introduction, Lenstra's (1989) algorithm gives a deterministic polynomial-time method to find a normal basis if an irreducible polynomial of degree n in $\mathbb{F}_q[x]$ is given.

5. Bounds on finite field functions

The Chebyshev ϑ function is defined for $x \geq 2$ by $\vartheta(x) = \sum_{p \leq x} \log_e p$, where the sum is extended over all primes $p \leq x$. From Rosser & Schoenfeld (1962), Theorem 4, we get the following bounds on ϑ.

FACT 5.1. *For $x \geq 563$,*

$$x\left(1 - \frac{1}{2\log_e x}\right) < \vartheta(x) < x\left(1 + \frac{1}{2\log_e x}\right).$$

We can define an analogue of the Chebyshev ϑ function for finite fields. For $x \geq 1$, let

$$\Theta_q(x) = \sum_{\substack{g \in I_q \\ |g| \leq x}} \deg g = \sum_{\substack{g \in I_q \\ |g| \leq x}} \log_q |g|,$$

where I_q is the set of monic irreducible polynomials of positive degree in $\mathbb{F}_q[x]$. $\Theta_q(x)$ remains constant when $q^b \leq x < q^{b+1}$, with $b \in \mathbb{N}$, and therefore we obtain sharp estimates for $\Theta_q(x)$ only when $x = q^b$ for some $b \in \mathbb{N}$.

THEOREM 5.2. *For any prime power q and $x = q^b$ for some integer $b \geq 1$, we have*

$$\frac{qx}{q-1} \cdot \left(1 - \frac{7}{2\sqrt{x}}\right) < \Theta_q(x) < \frac{qx}{q-1}.$$

For any real $x \geq q$, we have

$$\frac{x}{q-1} \cdot \left(1 - \frac{7\sqrt{q}}{2\sqrt{x}}\right) < \Theta_q(x) < \frac{qx}{q-1}.$$

PROOF. Let $x = q^b$ for some integer $b \geq 1$. By definition,

$$\Theta_q(x) = \sum_{\substack{g \in I_q \\ |g| \leq x}} \deg g = \sum_{1 \leq d \leq b} dN_q(d).$$

From the bounds on N_q given in equation (3.1) we get

$$\Theta_q(x) \leq \sum_{1 \leq d \leq b} q^d = \frac{q^{1+b}}{q-1} \cdot \left(1 - \frac{1}{q^{1+b}}\right) = \frac{qx}{q-1} \cdot \left(1 - \frac{1}{qx}\right) < \frac{qx}{q-1},$$

$$\begin{aligned}
\Theta_q(x) &\geq \sum_{1 \leq d \leq b} q^d - \frac{q}{q-1} \cdot \sum_{1 \leq d \leq b} q^{d/2} \\
&= \frac{qx}{q-1} \cdot \left(1 - \frac{1}{qx}\right) - \frac{q}{q-1} \cdot \frac{\sqrt{qx}}{\sqrt{q}-1} \cdot \left(1 - \frac{1}{\sqrt{qx}}\right) \\
&= \frac{qx}{q-1} - \frac{1}{q-1} - \frac{q}{q-1} \cdot \frac{\sqrt{qx}}{\sqrt{q}-1} + \frac{q}{(q-1)(\sqrt{q}-1)} \\
&> \frac{qx}{q-1} - \frac{q\sqrt{x}}{q-1} \cdot \frac{\sqrt{q}}{\sqrt{q}-1} > \frac{qx}{q-1} \cdot \left(1 - \frac{7}{2\sqrt{x}}\right),
\end{aligned}$$

since $q \geq 2$. For an arbitrary real $x \geq q$, let $x_0 = q^{\lfloor \log_q x \rfloor} \leq x$. Then $\Theta_q(x) = \Theta_q(x_0)$, and since the upper bound on Θ_q for powers of q is an increasing function on $\mathbb{R}$, it holds for any $x \geq q$. The lower bound is an increasing function of x for $x > 49/16$. Using the fact that $x/q < x_0$, when $x/q > 49/16$ the claimed lower bound follows. For $q < x \leq 49x/16$ the claim is easily verified. □

We also make use of the prime number function $\pi(x) = \#\{p \leq x \mid p \text{ prime}\}$. Rosser & Schoenfeld (1962), Theorem 1, gives the following bounds on π.

FACT 5.3. *For any $x \geq 17$,*

$$\frac{x}{\log_e x} < \pi(x) < \frac{x}{\log_e x} \cdot \left(1 + \frac{3}{2 \log_e x}\right).$$

The analogue of the number-theoretic function π for a finite field $\mathbb{F}_q$ is $\Pi_q(x) = \#\{g \in I_q : |g| \leq x\}$, for any prime power q and $x \geq 1$. We will make use of the following function in the analysis of Π_q:

$$\gamma_q(x) = \sum_{1 \leq d \leq \log_q x} \frac{q^d}{d}$$

for $q \geq 2$ and $x \geq q$. It can be bounded from above as follows:

LEMMA 5.4. *For any $x \geq q$ and $q \geq 2$,*

$$\gamma_q(x) \leq \frac{q}{q-1} \cdot \frac{x}{\log_q x} \cdot \left(1 + \frac{2}{\log_q x}\right)$$

PROOF. First, suppose $x = q^b$ for some integer $b \geq 1$. For $1 \leq b \leq 6$ the claim can be verified directly by considering each side of the inequality as a function of q alone. We omit the details. We prove the claim by induction on $b \geq 7$.

$$\begin{aligned}\sum_{1 \leq d \leq b} \frac{q^d}{d} &= \frac{q^b}{b} + \sum_{1 \leq d \leq b-1} \frac{q^d}{d} \leq \frac{q^b}{b} + \frac{q}{q-1} \cdot \frac{q^{b-1}}{b-1} + \frac{2q}{q-1} \cdot \frac{q^{b-1}}{(b-1)^2} \\ &= \frac{q}{q-1} \frac{q^b}{b} + \frac{q}{q-1} \cdot \frac{q^b}{b^2} R(q,b)\end{aligned}$$

where

$$R(q,b) = \frac{b}{(b-1)q} + \frac{2b^2}{(b-1)^2 q}.$$

Since $R(q,b)$ is a decreasing function of q and b, $R(q,b) \leq R(2,7) = 35/18 < 2$, so the claim is true for $q \geq 2$ and integer $b \geq 1$.

For any real $x \geq q$, let $x_0 = q^{\lfloor \log_q x \rfloor} \leq x$. Note first that

$$\frac{q}{q-1} \cdot \frac{x}{\log_q x} \cdot \left(1 + \frac{2}{\log_q x}\right)$$

is an increasing function of x for $x \geq q^3$, so the claim, already proven for x_0, holds for all real $x \geq q^3$. It is easily verified that the claim holds for all real x with $q < x < q^3$ as well. □

We will now show an analogue of the prime number theorem for $\mathbb{F}_q[x]$. $\Pi_q(x)$ remains constant when $q^b \leq x < q^{b+1}$, with $b \in \mathbb{N}$, and therefore we obtain sharp estimates for $\Pi_q(x)$ only when $x = q^b$ for some $b \in \mathbb{N}$.

THEOREM 5.5. *For any prime power q, and $x = q^b$ for some integer $b \geq 1$,*

$$\frac{q}{q-1} \cdot \frac{x}{\log_q x} \cdot \left(1 - \frac{7}{2\sqrt{x}}\right) < \Pi_q(x) < \frac{q}{q-1} \cdot \frac{x}{\log_q x} \cdot \left(1 + \frac{2}{\log_q x}\right).$$

For any real $x \geq q$, we have

$$\frac{1}{q-1} \cdot \frac{x}{\log_q x} \cdot \left(1 - \frac{7\sqrt{q}}{2\sqrt{x}}\right) < \Pi_q(x) < \frac{q}{q-1} \cdot \frac{x}{\log_q x} \cdot \left(1 + \frac{2}{\log_q x}\right).$$

PROOF. The lower bound follows from

$$\Pi_q(x) \log_q x = \sum_{\substack{g \in I_q \\ |g| \leq x}} \log_q x \geq \sum_{\substack{g \in I_q \\ |g| \leq x}} \deg g = \Theta_q(x),$$

and the lower bounds in Theorem 5.2. For the upper bound, note that for arbitrary real $x \geq q \geq 2$,

$$\Pi_q(x) = \sum_{1 \leq d \leq \log_q x} N_q(d) \leq \sum_{1 \leq d \leq \log_q x} \frac{q^d}{d} \leq \gamma_q(x)$$

by (3.1) and Lemma 5.4 proves the claim. $\square$

The function $\gamma_q(x)$ is a better approximation to $\Pi_q(x)$, with error only $O(\sqrt{x})$.

For $n \in \mathbb{N}$, let $\delta(n) = \#\{d \in \mathbb{N}, d|n, d \text{ is squarefree}\} = 2^{\omega(n)}$, where $\omega(n)$ is the number of primes dividing n. Hardy & Wright (1962), Section 22.11 show the "normal order" of $\omega(n)$ is $\log \log n$. We can bound δ from above as follows.

THEOREM 5.6. *For $n \geq 9$,*

$$\log_2 \delta(n) \leq \frac{\log_e n}{\log_e \log_e n} \cdot \left(1 + \frac{1.25 \log_e \log_e \log_e n}{\log_e \log_e n}\right).$$

PROOF. First, note that $\log_2 \delta(n)$ is the number of distinct prime divisors of n. Suppose $n = q_1^{e_1} q_2^{e_2} \cdots q_k^{e_k}$ where $k, e_1, e_2, \ldots, e_k$ are positive integers and $q_1 < q_2 < \cdots < q_k$ are primes. If $n_0 = q_1 q_2 \cdots q_k$, then $\delta(n_0) = \delta(n)$, while $n_0 \leq n$. Since we are trying to prove that δ is bounded above by an increasing function of n for $n > e$, if the claim fails for n, it also fails for n_0 as well. We can therefore assume without loss of generality that $e_1 = e_2 = \cdots = e_k = 1$. Now suppose p is a prime less than q_k and $p \notin \{q_1, \ldots, q_{k-1}\}$. Let $n_1 = p q_1 q_2 \cdots q_{k-1}$. Once again $\delta(n_1) = \delta(n)$, and if the claim fails for n, it also fails for $n_1 < n$. Thus, we can assume without loss of generality that n is the product of all primes less than some $x \geq 2$.

Let $x \geq 3989$ and let n be the product of all primes less than or equal to x (the number 3989 is chosen only for convenience in this proof). Then

$$\log_2 \delta(n) = \pi(x) < \frac{x}{\log_e x} \cdot \left(1 + \frac{3}{2 \log_e x}\right)$$

by Fact 5.3. From Fact 5.1, $\log_e n = \vartheta(x) < x(1 + 1/(2 \log_e x)) < 1.1x$ or $x > 0.9 \log_e n$, for $x \geq 3989$ or $n \geq \exp(\vartheta(3989))$. Also by Fact 5.1,

$$\log_e n = \vartheta(x) > x\left(1 - \frac{1}{2 \log_e x}\right) > x\left(1 - \frac{1}{2 \log_e \log_e n + 2 \log_e 0.9}\right),$$

so that

$$x < \left(1 - \frac{1}{2\log_e \log_e n + 2\log_e 0.9}\right)^{-1} \log_e n < (1+u)\log_e n,$$

where $u = \log_e \log_e \log_e n/(2\log_e \log_e n) > 0$, since $x \geq 3989$ or $n \geq \exp(\vartheta(3989))$. Substituting this upper bound on x into our upper bound for $\log_2 \delta(n)$ in terms of x, we get

$$\begin{aligned}
\log_2 \delta(n) &< \frac{\log_e n}{\log_e((1+u)\log_e n)} \cdot (1+u)\left(1 + \frac{3}{2\log_e((1+u)\log_e n)}\right) \\
&< \frac{\log_e n}{\log_e \log_e n} \cdot (1+u)\left(1 + \frac{3}{2\log_e \log_e n}\right) \\
&< \frac{\log_e n}{\log_e \log_e n} \cdot \left(1 + \frac{1.25\log_e \log_e \log_e n}{\log_e \log_e n}\right)
\end{aligned}$$

for $n > \exp(\vartheta(3989))$.

For $i \geq 1$, let P_i be the product of the i smallest primes. The number 3989 is the 550th smallest prime, so the claim holds for $n \geq P_{550}$. Using the computer algebra program MAPLE 4.3, we verified that the claim holds for P_i, where $3 \leq i < 550$, and for $9 \leq n < P_3 = 30$. This shows the claim is true for all integers $n \geq 9$. □

The inequality of this theorem is false for $n = 8$.

For polynomials over finite fields, the analogue of δ is as follows. For any prime power q, and any $f \in \mathbb{F}_q[x]$, let $\Delta_q(f)$ be the number of distinct, monic, squarefree divisors of f (including the divisor 1). We bound Δ_q from above in the next theorem, and show that this bound cannot be improved much.

Theorem 5.7. *Let $f \in \mathbb{F}_q[x]$ of degree n. For $n > 1$, $\log_2 \Delta_q(f) \leq n$. For $n > q$,*

$$\log_2 \Delta_q(f) < \frac{n}{\log_q n} \cdot \left(1 + \frac{3.5\log_e \log_e n}{\log_e n}\right).$$

Furthermore, for any fixed prime power q, there exist an infinite number of $f \in \mathbb{F}_q[x]$ such that

$$\log_2 \Delta_q(f) \geq \frac{n}{\log_q n} \cdot \left(1 - \frac{5}{\log_q n}\right)$$

where $n = \deg f$.

Proof. We begin with the upper bound. First note that $\log_2 \Delta_q(f)$ is the number of distinct divisors of f in I_q. Suppose $f = g_1^{e_1} g_2^{e_2} \cdots g_k^{e_k}$ where $k, e_1, e_2, \ldots, e_k$ are positive integers and $g_1, g_2, \ldots, g_k \in I_q$ are pairwise distinct. If $f_0 = g_1 g_2 \cdots g_k$, then $\Delta_q(f) = \Delta_q(f_0)$, while $n_0 = \deg f_0 \leq n$. Since we are trying to prove $\log_2 \Delta_q(f)$ is bounded above above by an increasing function of n for $n \geq e$, if the claim fails for f, it also fails for f_0. We can therefore assume without loss of generality that $e_1 = \cdots = e_k = 1$. Now suppose $\deg g_1 \leq \deg g_2 \leq \cdots \leq \deg g_k$ and $h \notin \{g_1, \ldots, g_k\}$ for some $h \in I_q$ with $\deg h < \deg g_k$, and let $f_1 = h g_1 \cdots g_{k-1}$. Once again $\Delta_q(f_1) = \Delta_q(f)$, and if the claim fails for f, it also fails for f_1 (because $\deg f_1 < \deg f$). Therefore, we can assume without loss of generality that, for all $h|f$ with $h \in I_q$, all $h_0 \in I_q$ with $\deg h_0 < \deg h$ also divide f. In other words, f is the product of the maximum number of distinct irreducible polynomials for its degree.

If $1 \le n \le q$, f is the product of linear polynomials in $\mathbb{F}_q[x]$. Thus, $\log_2 \Delta_q(f) = n$. If $n > q$, we write $\log_2 \Delta_q(f) = P_1 + P_2$, where $f = g_1 \cdots g_k$,

$$P_1 = \sum_{\substack{|g_i| \le \beta \\ 1 \le i \le k}} 1, \qquad P_2 = \sum_{\substack{|g_i| > \beta \\ 1 \le i \le k}} 1,$$

and $\beta = n/\log_e n$. If $\beta < q$, then $P_1 = 0$. For $\beta > q$,

$$P_1 = \Pi_q(\beta) \le \frac{q\beta}{(q-1)\log_q \beta} \cdot \left(1 + \frac{2}{\log_q \beta}\right),$$

by Theorem 5.5. To bound P_2, we note that

$$q^n = |f| = \prod_{1 \le i \le k} |g_i| \ge \prod_{\substack{|g_i| > \beta \\ 1 \le i \le k}} \beta = \beta^{P_2},$$

or $P_2 \le n/\log_q \beta$. We set $\lambda_1 = \log_e n$ and $\lambda_2 = \log_e \log_e n$. If $\beta < q$, this gives

$$\begin{aligned} \log_2 \Delta_q(f) = P_2 &< \frac{n}{\log_q \beta} = \frac{n}{\log_q n} \cdot \left(\frac{\lambda_1}{\lambda_1 - \lambda_2}\right) \\ &= \frac{n}{\log_q n} \cdot \left(1 + \frac{\lambda_2}{\lambda_1 - \lambda_2}\right) \le \frac{n}{\log_q n} \cdot \left(1 + \frac{1.4\lambda_2}{\lambda_1}\right) \end{aligned}$$

for $n \ge 1619 > e^{e^2}$, using the fact that $1/(\lambda_1 - \lambda_2) \le 1.38/\lambda_1$ for $n \ge 1619$. If $\beta \ge q$,

$$\begin{aligned} \log_2 \Delta_q(f) &< \frac{n}{\log_q \beta} + \frac{q\beta}{(q-1)\log_q \beta} \cdot \left(1 + \frac{2}{\log_q \beta}\right) \\ &= \frac{n}{\log_q n} \cdot \left(\frac{\lambda_1}{\log_e \beta} + \frac{q\lambda_1}{(q-1)n\log_e \beta} \cdot \left(1 + \frac{2\log_e q}{\log_e \beta}\right)\right) \\ &= \frac{n}{\log_q n} \cdot \left(\frac{\lambda_1 + h(q,n)}{\lambda_1 - \lambda_2}\right) = \frac{n}{\log_q n} \cdot \left(1 + \frac{h(q,n) + \lambda_2}{\lambda_1 - \lambda_2}\right), \end{aligned}$$

where

$$h(q,n) = \frac{q}{q-1} \cdot \left(1 + \frac{2\log_e q}{\lambda_1 - \lambda_2}\right).$$

For $n \ge 1619$, $h(q,n)$ is a decreasing function of n. Also, for $n \ge 1619$, $h(q,n)$ is an increasing function of q for $q \ge 7$, and achieves its maximum with $q \ge 7$. To maximize h, we choose q as large as possible. Since $\beta = n/\lambda_1 > q$, we have $h(q,n) < h(n/\lambda_1, n) = 3n/(n - \lambda_1) \le 3.02$ for $n \ge 1619$. It follows that

$$\begin{aligned} \log_2 \Delta_q(f) &< \frac{n}{\log_q n} \cdot \left(1 + \frac{3.02 + \lambda_2}{\lambda_1 - \lambda_2}\right) < \frac{n}{\log_q n} \left(1 + \frac{2.51\lambda_2}{\lambda_1 - \lambda_2}\right) \\ &< \frac{n}{\log_q n} \left(1 + \frac{3.5\lambda_2}{\lambda_1}\right) \end{aligned}$$

for $n \ge 1619$. For $q < n < 1619$, we verified the claim using the computer algebra system Maple 4.3.

To show that this upper bound cannot be improved much, let q be a fixed prime power, $x = q^m$ for some integer $m \geq 2$, and

$$f = \prod_{\substack{g \in I_q \\ |g| \leq x}} g.$$

Then

$$\log_2 \Delta_q(f) = \Pi_q(x) > \frac{qx}{(q-1)\log_q x} \cdot \left(1 - \frac{7}{2\sqrt{x}}\right),$$

by Theorem 5.5. This lower bound on $\Pi_q(x)$ is an increasing function of x for $x \geq 4$. The degree n of f satisfies

$$n = \sum_{\substack{g \in I_q \\ |g| \leq x}} \deg g = \Theta_q(x) < \frac{qx}{q-1}$$

by Theorem 5.2, which can be rewritten $x > n + n/q$. This shows

$$\begin{aligned}
\log_2 \Delta_q(f) &\geq \frac{n}{\log_q(n + n/q)} \cdot \left(1 - \frac{7}{2\sqrt{n + n/q}}\right) \\
&= \frac{n}{\log_q n}\left(1 - \frac{\log_q(1 + q^{-1})}{\log_q(n + n/q)}\right)\left(1 - \frac{7}{2\sqrt{n + n/q}}\right) \\
&\geq \frac{n}{\log_q n}\left(1 - \frac{1}{\log_q n} - \frac{7}{2\sqrt{n}}\right) \geq \frac{n}{\log_q n}\left(1 - \frac{5}{\log_q n}\right)
\end{aligned}$$

using the fact that $7/(2\sqrt{n}) < 4/\log_q n$ for all $n \geq q \geq 2$. □

6. Finding primitive normal elements

Let q be a prime power and n a positive integer. The multiplicative group of $\mathbb{F}_{q^n}$ is cyclic of order $q^n - 1$ (see Lidl & Niederreiter 1983, Theorem 2.8) and for any nonzero $\alpha \in \mathbb{F}_{q^n}$, we define the multiplicative order as $\mathrm{ord}(\alpha) = \min\{d \in \mathbb{N},\ d \geq 1,\ \alpha^d = 1\}$. An element of order $q^n - 1$ is called *primitive* and we denote by $\mathcal{P}$ the set of all these. It is well known that there are $\phi(q^n - 1)$ primitive elements in $\mathbb{F}_{q^n}$. There is no known general way to either generate or certify a primitive element in (probabilistic) polynomial time.

Let $\mathcal{N}$ be the set of normal elements in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. What is the probability that a randomly chosen element $\alpha \in \mathbb{F}_{q^n}$ is simultaneously primitive and normal over $\mathbb{F}_q$? This question was first addressed by Carlitz (1952) who showed in his statement (4.7) that

$$\varrho = \frac{|\mathcal{P} \cap \mathcal{N}|}{q^n} \geq \frac{\phi(q^n - 1) \cdot \Phi_q(x^n - 1)}{q^{2n}} - \frac{\delta(q^n - 1)\Delta_q(x^n - 1)}{q^{n/2}}. \qquad (6.1)$$

We will refer to the right hand side of this inequality as the Carlitz bound on the number of primitive normal elements. For sufficiently large fields $\mathbb{F}_{q^n}$, this tends towards $|\mathcal{P}|\cdot|\mathcal{N}|/q^{2n}$. It was later shown by Davenport (1968) that for q prime and $n \geq 2$ there exists a primitive normal element in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. Lenstra & Schoof (1987) showed that for all prime powers q and $n \geq 2$ that there exists a primitive normal element in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. We give a positive lower bound for ϱ for all but a finite number of pairs (q, n).

THEOREM 6.1. *Let q be a prime power, $n \geq 2$, and ϱ the probability that an element is primitive and normal in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. Then*

(i) if $n \geq 300$ and $n \geq q^4$, then $\varrho > 0.03/(\log_q n \cdot \log_e(n \log_e q))$,

(ii) if $n \geq 300$ and $n < q^4$, then $\varrho > 0.01/\log_e(n \log_e q)$,

(iii) if $300 \geq n \geq 2$ and $q \geq 2 \times 10^7$, then $\varrho > 0.003/\log_e(n \log_e q)$.

PROOF. Since our bounds on Φ_q and Δ_q from Theorems 3.4 and 5.7 depend upon the relationship between q and n, our lower bound on ϱ must be divided into a number of cases. First we examine the bounds for $\delta(q^n - 1)$ and $\phi(q^n - 1)$, which are valid when $q^n - 1 \geq 16$. We abbreviate $\lambda_1 = \log_e(q^n - 1)$, $\lambda_2 = \log_e \lambda_1$ and $\lambda_3 = \log_e \lambda_2$. By Theorem 5.6,

$$\log_2 \delta(q^n - 1) < \frac{\lambda_1}{\lambda_2} \cdot \left(1 + \frac{1.25\lambda_3}{\lambda_2}\right) < \frac{n \log_e q}{\lambda_2} \cdot \left(1 + \frac{1.25\lambda_3}{\lambda_2}\right),$$

since our upper bound on δ is an increasing function. This gives $\log_q \delta(q^n - 1) < r \cdot n/\lambda_2$ where $r = (1 + 1.25\lambda_3/\lambda_2) \cdot \log_e 2$. By Fact 3.5, we have $\phi(q^n - 1) > s \cdot q^n/\lambda_2$, where $s = c \cdot (1 - 1.41/\lambda_2^2) \cdot (1 - q^{-n})$. Applying (6.1) we have

$$\varrho > (s/\lambda_2) \cdot (\Phi_q(x^n - 1)/q^n) - q^{rn/\lambda_2 + \log_q \Delta_q(x^n-1) - n/2}. \tag{6.2}$$

By Theorem 5.7, for $n \geq q$ and $n \geq 300$ we have

$$\log_q \Delta_q(x^n - 1) < \frac{n \log_e 2}{\log_e n} \cdot \left(1 + \frac{3.5 \log_e \log_e n}{\log_e n}\right) < \frac{1.44n}{\log_e n}. \tag{6.3}$$

We will first look at case (i), where $n \geq 300$ and $n \geq q^4$. Here, $\Phi_q(x^n - 1) \geq q^n/(16 \log_q n)$ by Theorem 3.4. In this case we have $r < 0.97$ and $s > 0.53$. Using (6.2), and (6.3) we find

$$\varrho \geq \frac{0.53}{16 \log_q n \cdot \lambda_2} - q^{n(0.97/\lambda_2 + 1.44/\log_e n - 1/2)} > \frac{0.033}{\log_q n \cdot \lambda_2} - q^{-0.066n} > \frac{0.03}{\log_q n \cdot \lambda_2}.$$

For case (ii), when $n \geq 300$ and $n < q^4$, we consider two subcases. First, suppose $q < n < q^4$ and $n \geq 300$, which implies $q \geq 5$. In this case $r < 0.95$, and $s \geq 0.54$, while $\Phi_q(x^n - 1) > q^n/34$, by Theorem 3.4. Using (6.2) and (6.3), we find

$$\varrho \geq \frac{0.54}{34\lambda_2} - q^{n(0.95/\lambda_2 - 1.44/\log_e n - 1/2)} > \frac{0.015}{\lambda_2} - q^{-0.09n} > \frac{0.01}{\lambda_2}.$$

For the case when $300 \leq n \leq q$, $\Phi_q(x^n - 1) \leq q^n/34$, and by Theorem 5.7, $\log_q \Delta_q(x^n - 1) < n \log_q 2$. Also, $r \leq 0.93$ and $s \geq 0.54$, giving

$$\varrho \geq \frac{0.54}{34\lambda_2} - q^{n(0.93/\lambda_2 + \log_q 2 - 1/2)} > \frac{0.015}{\lambda_2} - q^{-0.25n} > \frac{0.01}{\lambda_2}.$$

Finally, for case (iii), we use the fact that for all $q \geq 2$ and $n \geq 2$, $\log_q \Delta_q(x^n - 1) \leq n \log_q 2$ by Theorem 5.7. Since $n < q$, $\Phi_q(x^n - 1) \geq q^n/34$ by Theorem 3.4. For $300 > n \geq 2$ and $q \geq 2 \times 10^7$, $r \leq 1.1003$ and $s \geq 0.496$, so that

$$\varrho \geq \frac{0.496}{34\lambda_2} - q^{n(1.003/\lambda_2 + \log_e 2/\log_e q - 1/2)} > \frac{0.014}{\lambda_2} - q^{-0.17n} > \frac{0.003}{\lambda_2}. \quad \square$$

This theorem covers all but finitely many values of (n, q). The exceptions, with $n < 300$ and $q < 2 \times 10^7$ are about 3.5×10^8 in number. The following proposition cuts down this number.

PROPOSITION 6.2. *If $9 \leq n < 300$ and $q \geq 300$, then $\varrho > 0.01/\log_e(n \log_e q)$.*

PROOF. We use the notation of Theorem 6.1. For $q \geq 300$ and $300 > n \geq 9$, we have $r \leq 1$ and $s \geq 0.5$. Also $\log_q \Delta_q(x^n - 1) \leq n \log_q 2$ by Theorem 5.7 and $\Phi_q(x^n - 1) > q^n/34$ by Theorem 3.4. Applying (6.2), this gives

$$\varrho \geq \frac{0.5}{34\lambda_2} - q^{n(1/\lambda_2 + \log_q 2 - 1/2)} > \frac{0.015}{\lambda_2} - q^{-0.25n} > \frac{0.01}{\lambda_2}. \quad \Box$$

This leaves us with about 8.3×10^6 exceptional pairs of fields. For each of these pairs we have used Maple 4.3 to compute the Carlitz bound (6.1), or at least a good lower bound for it. We found that it is at least 1/200 for all but 121 of the exceptional pairs of fields, and for the 121 remaining pairs it is negative. Thus, for all but 121 of the exceptional field pairs, the probability of finding a primitive normal element is at least 1/200. We have $q \leq 2729$ and $n \leq 21$ for all 121 remaining cases. Note also that the existence proofs of Davenport (1968) and Lenstra & Schoof (1987) showing $\varrho > 0$ both consider a (smaller) number of special cases.

COROLLARY 6.3. *Let q be a prime power and $n \geq 2$. There exists a probabilistic polynomial-time reduction from the problem of finding a primitive normal element in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ to finding a primitive element in $\mathbb{F}_{q^n}$.*

PROOF. Construct a lookup table of the (finitely many) exceptional cases not covered by Theorem 6.1 (and Proposition 6.2 and the following comments), mapping a pair (q, n) to a primitive normal element in $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. For a given input (q, n), where q is a prime power and $n \geq 2$, check if (q, n) is in the table of exceptions. If it is, return the primitive element stored there. If not, find a primitive element $\beta \in \mathbb{F}_{q^n}$. Randomly select an integer j between 1 and $q^n - 2$, compute $\alpha = \beta^j$, and test β for primitivity and normality over $\mathbb{F}_q$. In the case of Theorem 6.1 (i), we require an expected number of at most $34 \log_e(n \log_e q) \cdot \log_q n$ random choices. For the remaining two cases of Theorem 6.1, we require an expected number of at most $334 \log_e(n \log_e q)$ random choices. To test primitivity, we need only check that $\gcd(q^n - 1, j) = 1$, which requires $O\tilde{~}(n \log q)$ bit operations, while to test normality requires $O\tilde{~}(n^2 \log_e q)$ operations in $\mathbb{F}_q$ by Theorem 2.1. $\Box$

Acknowledgment

We thank an anonymous referee for pointing out several references, and an improvement in Proposition 2.4.

References

L. Adleman and H. W. Lenstra. Finding irreducible polynomials over finite fields. In *Proc. 18th Ann. ACM Symp. Theory of Computing*, pp. 350–355, Berkeley CA, 1986.

G. B. Agnew, R. C. Mullin, and S. A. Vanstone. Fast exponentiation in $GF(2^n)$. In *Advances in Cryptology—EUROCRYPT '88*, ed. C. G. Günther, vol. 330 of *Lecture Notes in Computer Science*, pp. 251–255. Springer (Berlin), 1988.

A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms.* Addison-Wesley (Reading MA), 1974.

A. A. Albert. *Fundamental Concepts of Higher Algebra.* University of Chicago Press (Chicago, Illinois), 1956.

T. M. Apostol. *Introduction to Analytic Number Theory.* Springer-Verlag (New York), 1976.

L. Babai, E. M. Luks, and Á. Seress. Fast management of permutation groups. In *Proc. 29th IEEE Symp. on Foundations of Computer Science*, pp. 272–282, White Plains, NY, 1988.

M. Ben-Or. Probabilistic algorithms in finite fields. In *Proc. 22nd IEEE Symp. Foundations Computer Science*, pp. 394–398, 1981.

T. Beth, B. M. Cook, and D. Gollmann. Architectures for exponentiation in $GF(2^n)$. In *Advances in Cryptology—CRYPTO '86*, ed. A. M. Odlyzko, vol. 263 of *Lecture Notes in Computer Science*, pp. 302–310. Springer (Berlin), 1986.

D. G. Cantor and E. Kaltofen. Fast multiplication of polynomials over arbitrary rings. Technical Report 87-35, Dept. of Computer Science, Rensselaer Polytechnic Institute, 1987. *Acta Inform.*, to appear.

L. Carlitz. Primitive roots in a finite field. *Trans. Amer. Math. Soc.* **73**, pp. 373–382, 1952.

D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. Symb. Comp.* **9**, pp. 251–280, 1990.

H. Davenport. Bases for finite fields. *J. London Math. Soc.* **43**, pp. 21–39, 1968.

G. Eisenstein. Lehrsätze. *J. reine angew. Math.* **39**, pp. 180–182, 1850.

J. von zur Gathen. Irreducibility of multivariate polynomials. *J. Computer System Sciences* **31**, pp. 225–264, 1985.

G. H. Hardy and E. M. Wright. *An Introduction to the Theory of Numbers.* Clarendon Press (Oxford), 1962.

K. Hensel. Ueber die Darstellung der Zahlen eines Gattungsbereiches für einen beliebigen Primdivisor. *J. Reine Angew. Math.* **103**, pp. 230–237, 1888.

M. A. Huang. Riemann hypothesis and finding roots over finite fields. In *Proc. 17th Ann. ACM Symp. Theory of Computing*, pp. 121–130, Providence RI, 1985.

B. A. Laws and C. K. Rushforth. A cellular-array multiplier for $GF(2^m)$. *IEEE Trans. Comput.* **C-20**, pp. 1573–1578, 1971.

H. W. Lenstra. Finding isomorphisms between finite fields. Manuscript, 1989.

H. W. Lenstra and R. J. Schoof. Primitive normal bases for finite fields. *Math. Comp.* **48**, pp. 217–231, 1987.

R. Lidl and H. Niederreiter. *Finite Fields*, vol. 20 of *Encyclopedia of Mathematics and its Applications.* Addison-Wesley (Reading MA), 1983.

O. Ore. Contributions to the theory of finite fields. *Trans. Amer. Math. Soc.* **36**, pp. 243–274, 1934.

M. O. Rabin. Probabilistic algorithms in finite fields. *SIAM J. Comp.* **9**, pp. 273–280, 1980.

J. B. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *Ill. J. Math.* **6**, pp. 64–94, 1962.

A. Schönhage and V. Strassen. Schnelle Multiplikation großer Zahlen. *Computing* **7**, pp. 281–292, 1971.

S. S. Schwarz. Construction of normal bases in cyclic extensions of a field. *Czechoslovak Math. Journal* **38**(113), pp. 291–312, 1988.

I. A. Semaev. Construction of polynomials irreducible over a finite field with linearly independent roots. *Math. USSR Sbornik* **63**(2), pp. 507–519, 1989.

V. Shoup. On the deterministic complexity of factoring polynomials over finite fields. *Information Processing Letters* **33**, pp. 261–267, 1990a.

V. Shoup. New algorithms for finding irreducible polynomials in finite fields. *Math. Comp.* **54**(189), pp. 435–447, January 1990b.

V. M. Sidel'nikov. On normal bases of a finite field. *Math. USSR Sbornik* **61**(2), pp. 485–494, 1988.

S. A. Stepanov and I. E. Shparlinsky. On structure complexity of normal basis of finite field. In *Fundamentals of Computation Theory, Proc.*, ed. L. Budach, R. G. Bukharajev, and O. B. Lupanov, vol. 278 of *Lecture Notes in Computer Science*, pp. 414–416. Springer (Berlin), 1987.

S.A. Stepanov and I.E. Shparlinsky. On the construction of a primitive normal basis of a finite field. *Mat. Sbornik* **180**, pp. 1067–1072, 1989.

D. R. Stinson. Some observations on parallel algorithms for fast exponentiation in $GF(2^n)$. *SIAM J. Comp.* **19**(4), pp. 711–717, August 1990.

B. L. van der Waerden. *Algebra, Erster Teil.* Springer-Verlag (Berlin), 7 edition, 1966.

C. C. Wang, T. K. Truong, H. M. Shao, L. J. Deutsch, J. K. Omura, and I. S. Reed. VLSI architectures for computing multiplications and inverses in $GF(2^m)$. *IEEE Trans. Comput.* **C-34**, pp. 709–717, 1985.