MATHEMATICS OF COMPUTATION
VOLUME 61, NUMBER 203
JULY 1993, PAGES 1-15

A SUBEXPONENTIAL ALGORITHM FOR DISCRETE LOGARITHMS OVER ALL FINITE FIELDS

LEONARD M. ADLEMAN AND JONATHAN DEMARRAIS

Dedicated to the memory of Professor Derrick Henry Lehmer

ABSTRACT. There are numerous subexponential algorithms for computing discrete logarithms over certain classes of finite fields. However, there appears to be no published subexponential algorithm for computing discrete logarithms over all finite fields. We present such an algorithm and a heuristic argument that there exists a $c \in \mathfrak{R}_{>0}$ such that for all sufficiently large prime powers p^n, the algorithm computes discrete logarithms over $\mathrm{GF}(p^n)$ within expected time: $e^{c(\log(p^n)\log\log(p^n))^{1/2}}$.

1. INTRODUCTION

Given α, β in a finite field, the discrete logarithm problem is to calculate an $x \in Z_{>0}$ (if such exists) such that

$$\alpha^x = \beta.$$

Interest in the discrete logarithm problem stems from the advent of public key cryptography, and with it the creation of cryptographic systems, which depend for their security on the difficulty of computing such logarithms (e.g., [10, 12]). While researchers have been successful in developing subexponential algorithms for computing discrete logarithms in finite fields of special form, no subexponential algorithm for computing discrete logarithms in *all* finite fields has emerged. We present such an algorithm along with a heuristic argument that there exists a $c \in \mathfrak{R}_{>0}$ such that for all sufficiently large prime powers p^n, the algorithm computes discrete logarithms over $\mathrm{GF}(p^n)$ within expected time:

$$e^{c(\log(p^n)\log\log(p^n))^{1/2}}.$$

There exist several algorithms which for all primes $p \in Z_{>0}$ compute discrete logarithms over $\mathrm{GF}(p)$ in time subexponential in p (e.g., [1, 15]). Further, for all primes $p \in Z_{>0}$, there exist algorithms which for all $n \in Z_{>0}$ compute discrete logarithms over $\mathrm{GF}(p^n)$ in time subexponential in p^n (for $p = 2$, this was first shown by Hellman and Reyneri [17] and improved by Coppersmith [8]; however, these approaches appear to generalize to an arbitrary prime p). Recently, Gordon [16] has announced that for all $n \in Z_{>0}$, there exists an algorithm which for all primes $p \in Z_{>0}$ computes discrete logarithms over $\mathrm{GF}(p^n)$

Received by the editor August 3, 1992 and, in revised form, December 3, 1992.
1991 *Mathematics Subject Classification*. Primary 11Y16, 11T71, 11Y40.

in time subexponential in p^n (the case $n = 2$ was previously established by ElGamal [13]). The previously most general subexponential algorithm appears to be that of Lovorn [21], which computes discrete logarithms in $\mathrm{GF}(p^n)$ for $\log(p) \leq n^{0.98}$.

Our subexponential method for all finite fields actually consists of two algorithms. They both may be described as "index calculus" methods [29, 23]. The first algorithm is for the case $n < p$. Here, $\mathrm{GF}(p^n)$ is represented by $O/(p)$, where O is a number ring and (p) is the prime ideal generated by p. An element of $O/(p)$ is considered "smooth" if and only if, when considered as an element of O, the ideal it generates factors into prime ideals of small norm. The second algorithm is for the case $n \geq p$. Here, $\mathrm{GF}(p^n)$ is represented by $(Z/pZ[x])/(f)$, where $f \in Z/pZ[x]$ is irreducible. An element of $(Z/pZ[x])/(f)$ is considered "smooth" if and only if, when considered as an element of $Z/pZ[x]$, it factors into irreducible polynomials of small degree.

While the second algorithm is rather "routine", an overview of the first algorithm may be useful. Consider computing the discrete logarithm of β with respect to the base α over $\mathrm{GF}(p)$, where p is prime. One can obtain a subexponential algorithm by representing $\mathrm{GF}(p)$ by Z/pZ and generating random integer pairs $\langle r, s\rangle$, calculating $\gamma \equiv \alpha^r \beta^s \bmod p$, and keeping the triple $\langle r, s, \gamma\rangle$ if and only if γ is B-smooth for an appropriate choice of B. When sufficiently many such good triples $\langle r_1, s_1, \gamma_1\rangle, \dots, \langle r_z, s_z, \gamma_z\rangle$ have been obtained, one can use linear algebra modulo $p-1$ to calculate $a_1, a_2, \dots, a_z \in Z_{\geq 0}^{<p-1}$ such that

$$\prod_{i=1}^{z} \gamma_i^{a_i} = \delta^{p-1}$$

for some integer δ, and hence that

$$\alpha^k \beta^l \equiv 1 \bmod p, \tag{1}$$

where $k = \sum_{i=1}^{z} a_i r_i$ and $l = \sum_{i=1}^{z} a_i s_i$. Generating such k, l pairs is tantamount to calculating the desired discrete logarithm.

Our first algorithm is a generalization of this approach to $\mathrm{GF}(p^n)$. By finding a number field of degree n over the rationals such that p is inert, $\mathrm{GF}(p^n)$ can be represented by O/pO, where O is the ring of integers in the number field. One can then proceed as before by generating random integer pairs $\langle r, s\rangle$, calculating $\gamma \equiv \alpha^s \beta^r \bmod p$, and keeping the triple $\langle r, s, \gamma\rangle$ if and only if γ is B-smooth for an appropriate choice of B. However, because O need not be a UFD, the notion of B-smoothness is generalized to mean that the ideal generated by γ is the product of prime ideals of small norm. Unfortunately, there are now two obstacles. First, γ will have an adequate chance of being B-smooth if and only if its absolute norm is small. We were only able to prove that this would be the case when the field in question was a subfield of a small-degree cyclotomic field. For this reason, cyclotomic polynomials and Gauss' theory of periods arise in the paper.

The second obstacle results from the linear algebra. We do not obtain $\prod_{i=1}^{z} \gamma_i^{a_i} = \delta^{p^n-1}$ for some (algebraic) integer δ as above. Rather, we obtain

$$\left(\prod_{i=1}^{z} \gamma_i^{a_i}\right) = I^{p^n-1}$$

for some ideal $I \subseteq O$. An algebraic integer, like $\prod_{i=1}^{z} \gamma_i^{a_i}$, which generates an ideal which is the $(p^n - 1)$st power of an ideal is called a $(p^n - 1)$-singular integer.

One can define two $(p^n - 1)$-singular integers to be equivalent if and only if their ratio is the $(p^n - 1)$st power of an element of the field. The equivalence classes form an Abelian group. The identity of this group is the class containing the $(p^n - 1)$st powers of algebraic integers. This group is generated by a small number h of elements (h depends on the structure of the ideal class group and the rank of the unit group in O). From this, the main virtue of singular integers follows: if h of them can be obtained, then there will exist a linear combination which is the $(p^n - 1)$st power of an algebraic integer. Thus, in the algorithm we will collect a number h of $(p^n - 1)$-singular integers $\gamma_1, \gamma_2, \dots, \gamma_h$, as above, and then find $b_1, b_2, \dots, b_h \in Z_{\geq 0}^{<p^n-1}$ such that

$$\prod_{i=1}^{h} \gamma_i^{b_i} = \delta^{p^n-1}$$

for some algebraic integer $\delta \in O$. From this, k and l as in equation (1) can be obtained in a straightforward way.

There remains the problem of calculating the $b_1, b_2, \dots, b_h$ described above. This is done with the device of "character signatures", which were introduced in the context of integer factoring [2]. The character signatures occurring in integer factoring are simpler than those occurring here, and a review of that setting may be rewarding.

2. Preliminaries

In this section some basic facts are presented.

Singular integers and character signatures. Here, some notions presented in [2] in the context of integer factoring are generalized.

Definition. For all number fields K with ring of integers O, for all $s \in Z_{>0}$, and for all $\sigma \in O$, σ is an s-singular integer (with respect to O) if and only if there exists an ideal $I \subseteq O$ such that $(\sigma) = I^s$.

Let K be a number field with ring of integers O, unit group E, and ideal class group C. Let $s \in Z_{>0}$, and let σ, τ be s-singular integers. Define $\sigma \approx \tau$ if and only if there exists $\alpha, \beta \in O$ such that $\alpha^s \sigma = \beta^s \tau$. Then $\approx$ is an equivalence relation on s-singular integers, and the set of equivalence classes forms a group $G(s)$ of exponents dividing s, with identity $I(s) = \{\alpha^s | \alpha \in O\}$ under the operation $[\alpha][\beta] \mapsto [\alpha\beta]$. There is a homomorphism ψ from $G(s)$ onto the group $C(s) = \{c | c \in C \;\&\; c^s = [(1)]\}$, $[\alpha] \overset{\psi}{\mapsto} [I]$, where $(\alpha) = I^s$.

The kernel of ψ is $\mathrm{Ker}(\psi) = \{[u] | u \in E\}$, and consequently $\mathrm{Ker}(\psi) \cong E/E^s$. Hence,

$$(*) \qquad G(s) \cong E/E^s \oplus C(s).$$

Definition. For all number fields K with ring of integers O, for all $s \in Z_{>0}$, for all prime ideals $P_1, P_2, \dots, P_z \subset O$, for all $l_1, l_2, \dots, l_z \in O$, and for all $\sigma \in O$: if for $i = 1, 2, \dots, z$, $(\sigma) + P_i = (1)$, $s|(N(P_i) - 1)$, and $l_i + P_i$ is a primitive sth root of unity in O/P_i^*, then the s-character signature of σ

with respect to $\langle P_1, l_1\rangle$, $\langle P_2, l_2\rangle, \dots, \langle P_z, l_z\rangle$ is $\langle e_1, e_2, \dots, e_z\rangle$, where for $i = 1, 2, \dots, z$, $\sigma^{(N(P_i)-1)/s} \equiv l_i^{e_i} \bmod P_i$ and $e_i \in Z_{\geq 0}^{<s}$.

Now assume that K is Abelian over Q; then it follows from the Čebotarev density theorem that for all $s \in Z_{>0}$, for all prime ideals $P_1, P_2, \dots, P_z \subset O$, and for all $c \in G(s)$, there exists a $\sigma \in O$ such that $[\sigma] = c$, and for $i = 1, 2, \dots, z$, $(\sigma) + P_i = (1)$. For $\langle P_1, l_1\rangle$, $\langle P_2, l_2\rangle, \dots, \langle P_z, l_z\rangle$ as above, let the map θ take c to the s-character signature of σ with respect to $\langle P_1, l_1\rangle, \langle P_2, l_2\rangle, \dots, \langle P_z, l_z\rangle$. The map θ is well defined on $G(s)$ and is a group homomorphism into $\bigoplus_{i=1}^{z} Z_s$.

Dependencies in Abelian groups. It is well documented how to find dependencies among elements of a vector space over a finite field. However, in Algorithms I and II, and many other factoring and discrete logarithm algorithms, it is necessary to find dependencies in modules over Z/mZ, where m is not prime. While in many papers this issue is taken for granted, we have included some of the relevant facts here. Readers may prefer to skip this exposition.

Theorem. *Let $p \in Z_{>0}$ be prime, and let $G = \bigoplus_{j=1}^{n} G_j$, where for $j = 1, 2, \dots, n$, G_j is cyclic of pth power order. Let $h_1, h_2, \dots, h_{n+1} \in G$. There exist $a_1, a_2, \dots, a_{n+1} \in Z$ such that* $\mathrm{GCD}(a_1, a_2, \dots, a_{n+1}) = 1$ *and* $\sum_{i=1}^{n+1} h_i a_i = 0$.

Proof. For $n = 1$, let g be a generator for G, and let $h_1 = x_1 g$ and $h_2 = x_2 g$. Then without loss of generality there exist $b_1, b_2 \in Z$ and $f \in Z_{\geq 0}$ such that $(b_1, p) = 1$, $x_1 = p^f b_1$, and $x_2 = p^f b_2$. Let $c \in Z$ be such that $cb_1 \equiv 1 \bmod p^e$, where p^e is the order of G, and let $a_1 = -cb_2$; then $a_1 h_1 + h_2 = 0$.

For $n > 1$, let g_j be a generator for G_j for $j = 1, 2, \dots, n$. For $i = 1, 2, \dots, n+1$, let

$$h_i = \sum_{j=1}^{n} e_{i,j} g_j.$$

Let $p^f \| \mathrm{GCD}(e_{1,1}, e_{2,1}, \dots, e_{n+1,1})$; then without loss of generality it can be assumed that $e_{1,1} = p^f a$, where $(a, p) = 1$. Consequently, for $i = 2, 3, \dots, n+1$, there exist $b_i \in Z$ such that

$$h_i' = h_i - b_i h_1 \in \bigoplus_{j=2}^{n} G_j.$$

By induction, there exist $a_2, a_3, \dots, a_{n+1} \in Z$ such that $\mathrm{GCD}(a_2, a_3, \dots, a_{n+1}) = 1$ and $\sum_{i=2}^{n+1} a_i h_i' = 0$. Let $a_1 = -\sum_{i=2}^{n+1} a_i b_i$; then $a_1, a_2, \dots, a_{n+1}$ are as desired. □

Corollary. *Let $n, s \in Z_{>0}$, and let G be a finite Abelian group of exponent dividing s such that $G = \bigoplus_{i=1}^{n} G_i$, where for $i = 1, 2, \dots, n$, G_i is cyclic. Let $h_1, h_2, \dots, h_{n+1} \in G$. There exist $a_1, a_2, \dots, a_{n+1} \in Z_{\geq 0}^{<s}$ such that* $\mathrm{GCD}(a_1, a_2, \dots, a_{n+1}) = 1$ *and* $\sum_{i=1}^{n+1} a_i h_i = 0$.

Proof. We have $G = \bigoplus G_p$, where G_p denotes the p-Sylow subgroup of G and the product is over all rational primes p. Applying the theorem for each $G_p \neq \{0\}$ and using the Chinese Remainder Theorem yields $b_1, b_2, \dots, b_{n+1} \in Z_{\geq 0}$ such that $\mathrm{GCD}(b_1, b_2, \dots, b_{n+1}, s) = 1$ and $\sum_{i=1}^{n+1} b_i h_i = 0$. For $i =$

$1, 2, \dots, n+1$, let $c_i \equiv b_i \bmod s$ and $c_i \in Z_{\geq 0}^{<s}$. Let $d = \mathrm{GCD}(c_1, c_2, \dots, c_{n+1})$. For $i = 1, 2, \dots, n+1$, let $a_i = c_i/d$. The $a_1, a_2, \dots, a_{n+1}$ are as desired. □

Subfields of cyclotomic fields. Let $q \in Z_{>0}$ be prime, and let $n|q-1$; then there exists a unique field $K_{q,n} \subseteq Q(\zeta_q)$, the qth cyclotomic field, such that $[K_{q,n} : Q] = n$. The following are well known [11]:

1. The ring of integers of $K_{q,n}$ is $O_{q,n} = Z[\eta_0, \eta_1, \dots, \eta_{n-1}]$, where for $i = 0, 1, \dots, n-1$, $\eta_i = \eta_{q,n,i} = \sum \zeta_q^a$, the sum being taken over the set of $a \in Z_{>0}^{\leq q-1}$ such that $\mathrm{ind}(a) \equiv i \bmod n$, where $\mathrm{ind}(a)$ denotes the index of a in Z/qZ^* with respect to a fixed generator.
2. $K_{q,n} = Q(\eta_0)$ (however, there exist q, n such that $O_{q,n} \neq Z[\eta_0]$).
3. The minimum polynomial for η_0 over Q is $f = f_{q,n} = \prod_{i=0}^{n-1}(x - \eta_i)$.
4. If $p \in Z_{>0}$ is prime and p is inert in $K_{q,n}$, then $O_{q,n}/(p)$ is a finite field with p^n elements and
$$R = R_{q,n,p} = \left\{ \sum_{i=0}^{n-1} a_i \eta_i | a_i \in Z_{\geq 0}^{<p},\ i = 0, 1, \dots, n-1 \right\}$$
is a complete set of representatives.

Arithmetic in $K_{q,n}$ may be done as follows (our description is essentially that of Edwards [11], which in turn is derived from Kummer).

Elements in $O_{q,n}$ will be represented in terms of the integer basis $\eta_0, \eta_1, \dots, \eta_{n-1}$.

First, for $i, j, k \in Z_{\geq 0}^{\leq n-1}$ calculate $c_{i,j,k} \in Z$ such that
$$\eta_i \eta_j = \sum_{k=0}^{n-1} c_{i,j,k} \eta_k ;$$
then multiplication in $O_{q,n}$ is straightforward.

Prime ideals of $O_{q,n}$ will be represented as follows. Let $s \neq q$ be a rational prime, and let f be the order of s in Z/qZ^*. Let $e = (q-1)/f$; then the splitting field of s is $K_{q,e}$. Let $g = (e, n)$; then s splits into g distinct prime ideals of residue class degree n/g in $O_{q,n}$.

Let $h \in Z/sZ[x]$ be an irreducible factor of $f_{q,q-1} = x^{q-1} + \dots + x + 1$ (the qth cyclotomic polynomial), and let σ be a generator for $\mathrm{GAL}(Q(\zeta_q)/Q)$ (the construction which follows produced the correct outcome for all choices).

For $i = 1, 2, \dots, g$, let $\tilde{S}_i \subseteq O_{q,q-1}$ be the prime ideal generated by s and $(h(\zeta_q))^{\sigma^i}$, and let $S_i = \tilde{S}_i \cap O_{q,n}$. Then $(s) = \prod_{i=1}^{g} S_i$ is the prime decomposition of s in $O_{q,n}$.

For $i = 1, 2, \dots, g$ and $j = 0, 1, \dots, e-1$, calculate $u_{i,j} \in Z_{\geq 0}^{<s}$ such that
$$u_{i,j} \equiv \eta_{q,e,j} \bmod \tilde{S}_i$$
(such $u_{i,j}$ always exist [11]). Let $U = \{u_{i,j} | j = 0, 1, \dots, e-1\}$ (U is the set of roots of $f_{q,e}$ mod s and is independent of i). Let
$$\psi_i = \prod_{j=0}^{e-1} \prod_{u \in U, u \neq u_{i,j}} (u - \eta_{i,j}).$$

For $i = 1, 2, \ldots, g$, $\langle s, \psi_i \rangle$ will represent the prime ideal S_i of $O_{q,n}$ lying above s.

Let $\alpha \in O_{q,n}$, and let $a \in Z_{\geq 0}$. Then

$$S_i^a|(\alpha) \quad \text{iff} \quad S_i^a O_{q,q-1}|\alpha O_{q,q-1} \quad \text{iff} \quad \widetilde{S}_i^a|\alpha O_{q,q-1} \quad \text{iff} \quad p^a|\psi_i^a \alpha.$$

The penultimate statement follows from Galois theory by noting that $\alpha \in K_{q,n}$. The last statement is essentially the first proposition of §4.10 in [11]. Hence, there is a computationally efficient method for determining the power of S_i which divides (α).

Next, consider singular integers and character signatures in $K_{q,n}$. Let $s \in Z_{>0}$. By Dirichlet's unit theorem, E/E^s can be written as the direct sum of at most n cyclic groups. Observing that the class number of $K_{q,n}$ is less than or equal to the class number of $Q(\zeta_q)$ [27, Theorem 10.1], which is less than or equal to q^{q^3} [22], it follows that $C(s)$ can be written as the direct sum of at most $q^3 \log_2(q)$ cyclic groups. By $(*)$ above, $G(s)$ can be written as the direct sum of at most $n + q^3 \log_2(q)$ cyclic groups. Let $H = n + q^3 \log_2(q) + 1$. By the preceding corollary, if $\sigma_1, \sigma_2, \ldots, \sigma_H$ are s-singular integers, then there exist $\delta \in O_{q,n}$ and $b_1, b_2, \ldots, b_H \in Z_{\geq 0}^{<s}$ such that $\text{GCD}(b_1, b_2, \ldots, b_H) = 1$ and $\prod_{j=1}^{H} \sigma_j^{b_j} = \delta^s$. Further, if $\theta_1 = \theta(\sigma_1)$, $\theta_2 = \theta(\sigma_2), \ldots, \theta_H = \theta(\sigma_H)$ are the s-signatures of $\sigma_1, \sigma_2, \ldots, \sigma_H$ with respect to some $\langle P_1, l_1 \rangle, \langle P_2, l_2 \rangle, \ldots, \langle P_z, l_z \rangle$, then $\sum_{j=1}^{H} b_j \theta_j = 0$. Finally, given the prime factorization of s, and given the s-signatures $\theta_1, \theta_2, \ldots, \theta_H$, the proofs of the preceding theorem and corollary give an algorithm to calculate a sequence of b_j's such that $\sum_{j=1}^{H} b_j \theta_j = 0$. This algorithm requires time at most $O(H^2 z \log^3(s))$.

Smooth numbers [7]. For all $\gamma \in \mathfrak{R}_{>0}^{\leq 1}$ and $\delta \in \mathfrak{R}_{>0}$, $L_x[\gamma, \delta]$ denotes the set of functions from $\mathfrak{R}$ to $\mathfrak{R}$ of the form

$$e^{(\delta + o(1))(\log(x))^{\gamma}(\log\log(x))^{1-\gamma}}, \qquad x \to \infty.$$

It will be helpful in the running time analyses which follow to note that for all $\gamma \in \mathfrak{R}_{>0}^{\leq 1}$, $\delta \in \mathfrak{R}_{>0}$, $L \in L_x[\gamma, \delta]$, and $c \in Z_{>0}$:

$$(\log(x)^c)L \in L_x[\gamma, \delta].$$

For all $\alpha, \gamma \in \mathfrak{R}_{>0}^{\leq 1}$ with $\alpha < \gamma$, for all $\beta, \delta \in \mathfrak{R}_{>0}$, $L_0 \in L_x[\gamma, \delta]$, and $L_1 \in L_x[\alpha, \beta]$, there exists an $L_2 \in L_x[\gamma - \alpha, (\gamma - \alpha)\delta/\beta]$ such that for all $N \in \mathfrak{R}_{>0}$, the probability that a positive integer less than or equal to $L_0(N)$ is $L_1(N)$-smooth (i.e., has all positive prime divisors less than or equal to $L_1(N)$) is at least $1/L_2(N)$.

Smooth polynomials. Algorithm II depends on finding polynomials over finite prime fields whose irreducible factors all have small degree. Call a polynomial m-smooth if and only if all of its irreducible factors have degree less than or equal to m. The following theorem gives a bound on the probability that a polynomial of degree n will be m-smooth. Our bound is not the best possible but is adequate for our purposes.

The following notation is generalized from Odlyzko [23].

Definition. For all $p, n, m \in Z_{>0}$ with p prime, let

$$N_p(n, m) = \#\{f | f \in Z/pZ[x] \;\&\; \text{degree}\, f = n \;\&\; f\ m\text{-smooth}\}.$$

Definition. For all $p, n, m \in Z_{>0}$ with p prime, let

$$P_p(n, m) = N_p(n, m)/N_p(n, n).$$

Theorem. *For all $p, n, m \in Z_{>0}$ with p prime and $n \geq m$, we have $P_p(n, m) \geq 1/(p^m n^{n/m})$.*

Proof. For all $k \in Z_{>0}$, let $S_k = \{f | f \in Z/pZ[x] \;\&\; \text{degree}\, f = k \;\&\; f$ monic and irreducible$\}$, and let $s_k = \#S_k$; then $(p^k - p^{k/2}\log(k))/k \leq s_k \leq p^k/k$ [25]. For all $k \in Z_{>0}$, let $T_k = \{f | f \in Z/pZ[x] \;\&\; \text{degree}\, f \leq k \;\&\; f$ monic and irreducible$\}$, and let $t_k = \#T_k$; then

$$\begin{aligned} t_k &= \sum_{i=1}^{k} s_i \geq \sum_{i=1}^{k} (p^i - p^{i/2}\log(i))/k \\ &= p^k/k + \sum_{i=1}^{k-1} (p^i - p^{(i+1)/2}\log(i+1))/k \geq p^k/k, \end{aligned}$$

since $p^i \geq p^{(i+1)/2}\log(i+1)$ for $i = 1, 2, \ldots, k-1$. Let r be the greatest integer less than n/m. Let $U = \{f | (\exists f_1, f_2, \ldots, f_r \in T_m)[f = \prod_{i=1}^{r} f_i]\}$, and let $u = \#U$. For all $f \in U$, we have $f \in N_p(n, m)$, thus $u \leq N_p(n, m)$. From probability (and the fact that $Z/pZ[x]$ is a UFD):

$$\begin{aligned} u &= \binom{t_m + r - 1}{r} \geq \binom{(p^m/m) + r - 1}{r} \\ &= ((p^m/m) + r - 1)!/((p^m/m) - 1)!r! \geq (p^m/mr)^r. \end{aligned}$$

Since $r \geq ((n+1)/m) - 1$, $p^{mr} \geq p^{n+1}/p^m$, and since $mr < n$, there holds $(mr)^r \leq n^{n/m}$. Hence, $(p^m/mr)^r \geq p^{n+1}/(p^m n^{n/m})$. Finally, since $N_p(n, n) = p^{n+1}$, we have $P_p(n, m) = N_p(n, m)/N_p(n, n) \geq 1/(p^m n^{n/m})$. □

Existence of a solution. It is possible that for $\alpha, \beta \in \text{GF}(p^n)$ with $\beta \neq 0$, the equation $\alpha^x = \beta$ will have no solution. However, for simplicity in the algorithms below, it will be assumed that α is a generator for $\text{GF}(p^n)^*$ and thus that a solution always exists. In the general case on inputs $\alpha, \beta \in \text{GF}(p^n)$, one may choose elements of $\text{GF}(p^n)$ at random until a generator γ is found and confirmed. Then use the algorithms below to calculate $x_1, x_2 \in Z_{\geq 0}^{<p^n - 1}$ such that $\gamma^{x_1} = \alpha$ and $\gamma^{x_2} = \beta$. The original problem can now be solved as follows: calculate $g_1 = (x_1, p^n - 1)$; if g_1 does not divide x_2, then there is no solution, else $x \equiv l(x_2/g_1) \bmod p^n - 1$, where $l \equiv (x_1/g_1)^{-1} \bmod((p^n - 1)/g_1)$. Since generators for $\text{GF}(p^n)^*$ are abundant [3, Lemma 4], finding one will require negligible time. Further, a candidate generator γ can be confirmed by first factoring $p^n - 1$ and establishing that for all primes $t | p^n - 1$, $\gamma^{(p^n - 1)/t} \neq 1$. Using an "$L[1/2, 1]$" factoring method (e.g., [19]), this process will add only negligible time to the algorithms below.

Notation. For all $p, n \in Z_{>0}$ with p prime, if we write $f \in Z/pZ[x]$, then it will be assumed that $f = \sum_{i=0}^{n} a_i x^i$, where for $i = 1, 2, \dots, n$, $a_i \in Z_{\geq 0}^{<p}$.

3. Algorithm I

This algorithm will be used for discrete logarithms over $\mathrm{GF}(p^n)$ when $p > n$.

Let $p \in Z_{>0}$ be prime and $f_1 \in Z/pZ[x]$ irreducible, monic of degree n. Then $(Z/pZ[x])/(f_1)$ is a finite field with p^n elements. Let $\alpha_1, \beta_1 \in Z/pZ[x]$ of degree less than n such that $[\alpha_1]$ generates $(Z/pZ[x])/(f_1)^*$ and $\beta_1 \not\equiv 0 \bmod f_1$. Hence, there exists an x such that $0 \leq x \leq p^n - 1$ and $\alpha_1^x \equiv \beta_1 \bmod f_1$. Assume that $p, f_1, \alpha_1, \beta_1$ are given and x is sought. Then one may proceed as follows.

As remarked in the introduction, it is necessary that we work in an nth-degree extension of the rationals which is contained in a cyclotomic field of small degree. For this reason, the original polynomial f_1 will be replaced with a new irreducible monic polynomial f such that $Q[x]/(f)$ is a field of the desired type.

Using the construction in [4], find an $f \in Z/pZ[x]$ irreducible of degree n in random time polynomial in $\log(p)$ and n (assuming ERH). By the construction in [4] (also see [6]), there exists a $\tilde{c} \in Z_{>0}$ such that $f = f_{q,n}$ for some prime $q \in Z_{>0}$ with $q \leq \tilde{c}n^4(\log(np))^2$ (assuming ERH). We have $(Z/pZ[x])/(f) \cong (Z/pZ[x])/(f_1)$. Using [18], calculate α_2 and $\beta_2 \in Z/pZ[x]$ of degree less than n such that $[\alpha_2]$ is the image of $[\alpha_1]$ and $[\beta_2]$ is the image of $[\beta_1]$ under this isomorphism. Hence, our original problem is reduced to the problem: given p, f, α_2, β_2 with $[\alpha_2]$ generating $(Z/pZ[x])/(f)^*$ and $\beta_2 \not\equiv 0 \bmod f$, calculate x such that $0 \leq x \leq p^n - 1$ and $\alpha_2^x \equiv \beta_2 \bmod f$.

Since f is irreducible in $Z/pZ[x]$, it follows that p is inert in $K_{q,n}$. There exists the following isomorphism from $(Z/pZ[x])/(f)$ to $O_{q,n}/(p)$:

$$\left[\sum_{i=0}^{n-1} g_i x^i\right] \mapsto \left[\sum_{i=0}^{n-1} g_i \left(\sum_{j=0}^{n-1} d_{i,j}\eta_{q,n,j}\right)\right],$$

where for $i = 0, 1, \dots, n-1$, $\eta_{q,n,0}^i = \sum_{j=0}^{n-1} d_{i,j}\eta_{q,n,j}$, with $d_{i,j} \in Z$.

Calculate $\alpha_3, \beta_3 \in O$ such that $[\alpha_3]$ is the image of $[\alpha_2]$ and $[\beta_3]$ is the image of $[\beta_2]$ under this isomorphism. By reducing coefficients modulo p, find $\alpha, \beta \in R_{q,n,p}$ such that $\alpha \equiv \alpha_3 \bmod p$ and $\beta \equiv \beta_3 \bmod p$. Hence, the original problem becomes that of calculating x such that $0 \leq x \leq p^n - 1$ and $\alpha^x \equiv \beta \bmod p$.

Below, a family of algorithms $\{A_y\}_{y \in Z_{>0}}$ is presented. It will be argued that for sufficiently large y: A_y on all inputs q, n, p, α, β such that $p, q \in Z_{>0}$ are prime, $n < p$, $n | q-1$, $q \leq \tilde{c}n^4(\log(np))^2$, p inert in $K_{q,n}$, and $\alpha, \beta \in R_{q,n,p}$ with $[\alpha]$ generating $O_{q,n}/(p)^*$ and $\beta \not\equiv 0 \bmod p$, outputs x such that $0 \leq x < p^n - 1$ and $\alpha^x \equiv \beta \bmod p$.

Let $L_0 \in L_x[1/2, \sqrt{1/2}]$.

Algorithm A_y.

Stage 0. Input q, n, p, α, β.

Stage 1. Set $N = p^{yn}$. Set (the "smoothness bound") $B = L_0(N)$. Set $H = n + q^3\log_2(q) + 1$.

Stage 2. Calculate $T = \{I | I$ is a prime ideal of O, $q \notin I$, and I lies over a rational prime $< B\}$. Let $w = \#T$, and let $\langle I_1, I_2, \ldots, I_w \rangle$ be an ordering of T.

Stage 3. Set $j = 1$. While $j \leq H$:

Stage 3(a). Set $z = 1$. While $z \leq w + 1$: Choose random r, s with $0 \leq r, s < p^n - 1$ and calculate $\gamma \in R_{q,n,p}$ such that $\gamma \equiv \alpha^r \beta^s \bmod(p)$. If $(\gamma) = \prod_{i=1}^{w} I_i^{e_i}$ (i.e., if the ideal generated by γ is B-smooth), then set $\gamma_{j,z} = \gamma$, $r_{j,z} = r$, $s_{j,z} = s$, $v_{j,z} = \langle e_1, e_2, \ldots, e_w \rangle$, and $z = z + 1$.

Stage 3(b). Calculate $a_1, a_2, \ldots, a_{w+1} \in Z_{\geq 0}^{<p^n - 1}$ such that $\mathrm{GCD}(a_1, a_2, \ldots, a_{w+1}) = 1$ and $\sum_{i=1}^{w+1} a_i v_{j,i} \equiv \langle 0, 0, \ldots, 0 \rangle \bmod p^n - 1$. Calculate $\sigma_j = \prod_{i=1}^{w+1} \gamma_{j,i}^{a_i}$. Set $j = j + 1$.

Stage 4. For $j = 1, 2, \ldots, H$, calculate θ_j the $(p^n - 1)$-signature of σ_j with respect to $\langle S_1, m_1 \rangle, \langle S_2, m_2 \rangle, \ldots, \langle S_{2H}, m_{2H} \rangle$, where for $j = 1, 2, \ldots, H$, $k = 1, 2, \ldots, 2H$, $S_k \subset O_{q,n}$ is a prime ideal such that $(\sigma_j) + S_k = (1)$, $(p^n - 1) | N(S_k) - 1$, and m_k is a primitive $(p^n - 1)$th root of unity in O/S_k.

Stage 5. Calculate $b_1, b_2, \ldots, b_H \in Z_{\geq 0}^{<p^n - 1}$ such that $\mathrm{GCD}(b_1, b_2, \ldots, b_H) = 1$ and $\sum_{j=1}^{H} b_j \theta_j \equiv \langle 0, 0, \ldots, 0 \rangle \bmod(p^n - 1)$.

Stage 6. Calculate $k = \sum_{j=1}^{H} \sum_{i=1}^{w+1} (r_{j,i} a_i b_j)$ and $l = \sum_{j=1}^{H} \sum_{i=1}^{w+1} (s_{j,i} a_i b_j)$. If $\alpha^k \beta^l \not\equiv 1 \bmod(p)$, then go to Stage 3.

Stage 7. If $(l, p^n - 1) \neq 1$, then go to Stage 3, else calculate and output $x \equiv -k/l \bmod p^n - 1$ and halt.

4. Analysis of Algorithm I

In this section computational details of Algorithm I will be described and there will be an analysis of the expected number of steps required by the algorithm on all inputs q, n, p, α, β such that $p, q \in Z_{>0}$ are prime with $n < p$, $n | q - 1$, $q \leq \tilde{c} n^4 (\log(np))^2$, p inert in $K_{q,n}$, and $\alpha, \beta \in R_{q,n,p}$ with $[\alpha]$ generating $O_{q,n}/(p)^*$ and $\beta \not\equiv 0 \bmod p$. For convenience, the argument will be for p^n sufficiently large.

To begin, consider the expected number of steps required by a single pass through each of the stages of the algorithm.

The time required for Stages 0, 1, 6, and 7 are dominated by the time required by other stages.

Stage 2: Test all numbers less than or equal to B for primality. For each prime $s \neq q$ found, calculate the representatives $\langle s, \psi_i \rangle$ of the prime ideals of $O_{q,n}$ lying above s and add them to T (see §2).

Using random polynomial-time primality testing [26, 3] and random polynomial-time finite field polynomial factorization [5], and observing that because of the size constraints on q, orders can be computed naively, it follows that there exists an $L_1 \in L_x[1/2, \sqrt{1/2}]$ such that the expected number of steps for a pass through Stage 2 is at most $L_1(N)$.

Further, since each rational prime has at most n primes lying over it in $O_{q,n}$, it follows that there exists an $L_2 \in L_x[1/2, \sqrt{1/2}]$ such that $w = \#T \leq L_2(N)$.

Stage 3(a): A γ will be tested for B-smoothness by the following method: First the norm of γ will be calculated and tested for B-smoothness. Those γ which have B-smooth norms will then be factored as ideals (see §2).

A bound on the norm of γ will be needed,

$$\gamma = \sum_{i=0}^{n-1} g_i \eta_i ,$$

where $0 \leq g_i \leq p-1$ for $i = 0, 1, \ldots, n-1$. Hence, γ is the sum of $q-1$ terms each of the form $g\zeta_q^c$, where $0 \leq g \leq p-1$ and $c \in Z_{\geq 0}^{<q}$. This is also the form of the n conjugates of γ. Hence, the norm of $\gamma = \prod_{\sigma \in \mathrm{Gal}(K_{q,n}/Q)} \gamma^\sigma$ is the sum of $(q-1)^n$ terms, the largest of which has absolute value p^n. By the constraints on q and n, it follows that there exists a $y_0 \in Z_{>0}$ such that $N(\gamma) \leq p^{y_0 n} \leq N$ for all algorithms A_y with $y \geq y_0$. Henceforth, assume that $y \geq y_0$.

Under the usual assumption [20] that the probability that $N(\gamma)$ is B-smooth (the exception of the prime q is inconsequential) is equal to the probability that a random positive integer less than N is B-smooth (see §2), there exists an $L_3 \in L_x[1/2, \sqrt{1/2}]$ such that the probability that γ is B-smooth (i.e., that all prime ideals dividing (γ) have norm less than or equal to B) is at least $1/L_3(N)$. Since w B-smooth γ's are needed, it follows that there exists an $L_4 \in L_x[1/2, \sqrt{2}]$ such that the expected number of γ's which must be generated and tested for B-smoothness is at most $L_4(N)$.

The norm of each γ may be tested for B-smoothness naively. Hence, there exists an $L_5 \in L_x[1/2, 3/\sqrt{2}]$ such that the expected number of steps required for a single pass through Stage 3(a) will be at most $L_5(N)$.

Stage 3(b): As indicated in §2, there must exist $a_1, a_2, \ldots, a_{w+1} \in Z_{\geq 0}^{<p^n-1}$ such that $\mathrm{GCD}(a_1, a_2, \ldots, a_{w+1}) = 1$ and $\sum_{i=1}^{w+1} a_i v_{j,i} \equiv \langle 0, 0, \ldots, 0\rangle \bmod (p^n-1)$. Further, as indicated in §2, there exists an algorithm which will find $a_1, a_2, \ldots, a_{w+1}$ in $O(w^3 \log^2(p^n))$ steps. Hence, there exists an $L_6 \in L_x[1/2, 3/\sqrt{2}]$ such that the expected time for a single pass through Stage 3(b) is at most $L_6(N)$.

Stage 4: Check numbers of the form $1+a(q(p^n-1))$ until primes $s_1, s_2, \ldots, s_{2H/n}$ are found. For $k = 1, 2, \ldots, 2H/n$, let $g_k \in Z_{\geq 0}^{<s_k}$ generate Z/s_kZ^* and let $g \in Z_{\geq 0}^{<q}$ generate Z/qZ^*. For $k = 1, 2, \ldots, 2H/n$, $l = 1, 2, \ldots, n$: Let $\widetilde{S}_{k,l} \subseteq O_{q,q-1}$ be the prime ideal generated by s and $\zeta_q^{d_l} - c_k$, where $c_k \equiv g_k^{a(p^n-1)} \bmod s$ and $d_l \equiv g^l \bmod q$. Let $S_{k,l} = \widetilde{S}_{k,l} \cap O_{q,n}$. Then $S_{k,1}, S_{k,2}, \ldots, S_{k,n}$ are the (distinct, residue class degree 1) prime ideals of $O_{q,n}$ lying above s_k. Since $s_k \equiv 1 \bmod q(p^n-1)$, it follows that $(p^n-1) | (N(S_{k,l})-1)$ and $N(S_{k,l}) > B$. Since for $j = 1, 2, \ldots, H$, (σ_j) is B-smooth, it follows that $(\sigma_j) + S_{k,l} = (1)$. Let $m_k = g_k^{aq} \bmod s_k$. Then the $2H$ pairs $\langle S_{k,l}, m_k\rangle$ will be as required for Stage 4.

Assume that approximately the "expected" number of primes will be found in an arithmetic progression: assume that for all $m, b \in Z_{>0}$, with $b > m\log(m)^3$: $\#\{a | 1+am < b \,\&\, 1+am \text{ prime}\} > b/m\log(b)^2$. If we let $v = 2H/n$ and $m = q(p^n-1)$, then all of the v primes needed above can be found by checking less than $v\log(v)^3\log(m)^3$ a's, and each prime s found will be less than $mv\log(v)^3\log(m)^3$. The constraints on n and q imply that there exists a $c_1, c_2 \in Z_{>0}$ such that $v\log(v)^3\log(m)^3 < (n\log(p))^{c_1}$ and $mv\log(v)^3\log(m)^3 < p^n(n\log(p))^{c_2}$. Hence, the required primes can be found and tested for primality [3, 26] in a negligible number of steps.

Generators for Z/s_kZ^* are abundant [3, Lemma 4]. Checking a candidate g to determine whether it is a generator will be done by factoring $s-1$ and testing that for all primes $t|s-1$, $g^{(s-1)/t} \not\equiv 1 \bmod s$. The factorization can be done using an "$L[1/2, 1]$" factoring method (e.g., [19]). A similar argument shows that a generator for Z/qZ^* can be found in a negligible number of steps.

We have $O_{q,n}/S_{k,l} \cong Z/s_kZ$, where the isomorphism is induced by $\zeta_q^{d_l} \mapsto c_k$. Hence, the calculations of the (p^n-1)-signatures of the σ_j's is a set of discrete logarithm problems over Z/s_kZ. Using the bounds on $2H$ and the primes s together with an "$L[1/2, 1]$" discrete logarithm algorithm for finite prime fields (e.g., [24]), we conclude that there exists an $L_7 \in L_x[1/2, 1]$ such that the expected number of steps required for a single pass through Stage 4 is at most $L_7(N)$.

Stage 5: By the analysis in §2, the required $b_1, b_2, \dots, b_H$ exist and can be found in time $O(H^3 \log^3(p^n-1))$. Using the bounds on q, we conclude that the number of steps required for a single pass through Stage 5 is negligible.

It will next be shown that the expected number of passes through stages of the algorithm is negligible. Stages will be repeated only if required in Stage 6 or Stage 7.

Stage 6 will cause stages of the algorithm to be repeated only if $\alpha^k\beta^l \not\equiv 1 \bmod (p)$. One has

$$\alpha^k\beta^l = \prod_{i,j} \alpha^{r_{j,i}a_ib_j}\beta^{s_{j,i}a_ib_j} = \prod_j \left(\prod_i (\alpha^{r_{j,i}}\beta^{s_{j,i}})^{a_i}\right)^{b_j} \equiv \prod_j \left(\prod_i \gamma_{j,i}^{a_i}\right)^{b_j} = \prod_j \sigma_j^{b_j}.$$

By construction, the σ_j are (p^n-1)-singular integers. By the arguments in §2 there exists a $\delta \in O_{q,n}$ and $b_1, b_2, \dots, b_H \in Z_{\geq 0}^{<p^n-1}$ such that $\mathrm{GCD}(b_1, b_2, \dots, b_H) = 1$ and $\prod_{j=1}^H \sigma_j^{b_j} = \delta^{p^n-1}$. Further, $G(p^n-1)$ is a group of indices dividing p^n-1, which is the direct product of at most $H-1$ cyclic groups (see §2). The signature homomorphism θ maps $G(p^n-1)$ into a group which is the direct product of $2H$ cyclic groups of order p^n-1. It is reasonable to assume that this map is an embedding, and hence that these $b_1, b_2, \dots, b_H$ are the ones found in Stage 5. It follows that

$$\alpha^k\beta^l \equiv \prod_j \sigma^{b_j} = \delta^{p^n-1} \equiv 1.$$

Stage 7 will cause stages of the algorithm to be repeated only if $(l, p^n-1) \neq 1$. However, $(l, p^n-1) = 1$ with probability $\phi(p^n-1)/(p^n-1) \geq 1/c \log p^n$, where $c \in \mathfrak{R}_{>0}$ is independent of p and n [3, Lemma 4]. Briefly, this can be argued as follows: Since from Stage 3(b), $\mathrm{GCD}(a_1, a_2, \dots, a_{w+1}) = 1$, and from Stage 5, $\mathrm{GCD}(b_1, b_2, \dots, b_H) = 1$, it follows that for all primes t dividing p^n-1, there exist $i \in Z_{>0}^{\leq w+1}$ and $j \in Z_{>0}^{\leq H}$ such that a_ib_j is relatively prime to t. Consider $\gamma_{j,i} \equiv \alpha^{r_{j,i}}\beta^{s_{j,i}}$, and observe that for all $s \in Z_{\geq 0}^{<p^n-1}$, there exists a unique $r \in Z_{\geq 0}^{<p^n-1}$ such that $\gamma_{j,i} \equiv \alpha^r\beta^s$. Hence, $s_{j,i}$ is "random" mod t and consequently $l = \sum_{j=1}^H \sum_{i=1}^{w+1}(s_{j,i}a_ib_j)$ is also "random" mod t.

Recalling that in Algorithm A_y we have $N = p^{yn}$, we may conclude that there exists a $c_I \in \mathfrak{R}_{>0}$ and an $L_I \in L_x[1/2, c_I]$ such that for all sufficiently large y, the expected number of steps required by Algorithm A_y on all inputs q, n, p, α, β such that $p, q \in Z_{>0}$ are prime, $n < p$, $n|q-1$, $q \le \tilde{c}n^4(\log(np))^2$, p inert in $K_{q,n}$, and $\alpha, \beta \in R_{q,n,p}$ with $[\alpha]$ generating $O_{q,n}/(p)^*$ and $\beta \not\equiv 0 \bmod p$ is $L_I(p^n)$. Hence, there exists a $c_I \in \mathfrak{R}_{>0}$ such that the expected number of steps required by Algorithm I (when $n < p$) is

$$e^{c_I(\log(p^n)\log\log(p^n))^{1/2}}.$$

Finally, it is clear from Stages 6 and 7 that the output of the algorithm is x such that $\alpha^x \equiv \beta \bmod p$.

5. Algorithm II

This algorithm will be used for discrete logarithms over $\mathrm{GF}(p^n)$ when $p \le n$.

Algorithm II is a generalization of the algorithm for $\mathrm{GF}(2^n)$ by Hellman and Reyneri discussed in Coppersmith [17, 8].

It is assumed that the inputs to the algorithm are p, f, α, β such that $p \in Z_{>0}$ is prime, $f \in Z/pZ[x]$ is monic, irreducible of degree $n \ge p$, and $\alpha, \beta \in Z/pZ[x]$ of degree less than n with $[\alpha] \in (Z/pZ[x])/(f)$ a generator of the multiplicative group and $\beta \not\equiv 0 \bmod f$.

Algorithm II.

Stage 0. Input f, p, α, β.

Stage 1. Set $n =$ degree of f, $m = \lceil (n\log(n)/\log(p))^{1/2} \rceil$.

Stage 2. Calculate $T = \{f_i | f_i \in Z/pZ[x],\ \deg(f_i) \le m,\ f_i$ irreducible and monic$\}$. Let $w = \#T$ and let $\langle f_1, f_2, \dots, f_w \rangle$ be an ordering of T.

Stage 3. Set $z = 1$. While $z \le w+1$: Choose random r, s with $0 \le r, s < p^n - 1$ and calculate $\gamma \in Z/pZ[x]$ of degree less than n such that $\gamma \equiv \alpha^r\beta^s \bmod f$. If $\gamma = \tilde{\gamma}\prod_{i=1}^{w} f_i^{e_i}$, where $\tilde{\gamma}$ is the leading coefficient of γ (i.e., if γ is m-smooth), then set $\gamma_z = \gamma$, $r_z = r$, $s_z = s$, $v_z = \langle e_1, e_2, \dots, e_w \rangle$, and $z = z+1$.

Stage 4. Calculate $a_1, a_2, \dots, a_{w+1} \in Z_{\ge 0}^{<p^n-1}$ such that $\mathrm{GCD}(a_1, a_2, \dots, a_{w+1}) = 1$ and $\sum_{i=1}^{w+1} a_i v_i \equiv \langle 0, 0, \dots, 0 \rangle \bmod (p^n - 1)$.

Stage 5. Calculate $k = \sum_{i=1}^{w+1}(r_i a_i)$ and $l = \sum_{i=1}^{w+1}(s_i a_i)$. Calculate $s \in Z_{>0}^{<p}$ such that $s \equiv \alpha^k\beta^l \bmod f$.

Stage 6. Calculate $y \in Z_{\ge 0}^{<p-1}$ such that $\alpha^{y((p^n-1)/(p-1))} \equiv s \bmod f$.

Stage 7. If $(l, p^n - 1) \ne 1$, then go to Stage 3, else calculate and output $x \equiv (y((p^n-1)/(p-1)) - k)/l \bmod p^n - 1$ and halt.

6. Analysis of Algorithm II

In this section the complexity of Algorithm II will be analyzed. For convenience it will be assumed that p^n is sufficiently large.

The time required for Stages 0, 1, 5, and 7 is dominated by the time required by other stages. Since $n \ge p$, it follows that the y required in Stage 6 can be found by exhaustion in a negligible amount of time.

Stage 2. Since every element in T is of degree at most m,

$$w < p^m \le e^{((n\log(n)/\log(p))^{1/2}+1)\log(p)} = e^{(n\log(n)\log(p))^{1/2}+\log(p)}$$
$$\le e^{(\log(p^n)\log\log(p^n))^{1/2}+\log(p)} \in L_{p^n}[1/2, 1]$$

(observe that $\log\log(p^n) > \log(n)$). Since irreducibility checking in $Z/pZ[x]$ can be done in time polynomial in n and $\log(p)$ [5], there exists an $L_1 \in$

$L_{p^n}[1/2, 1]$ such that all irreducible polynomials of degree less than or equal to m can be found by exhaustion within time L_1.

Stage 3. By choosing a random $0 \le r < p^n - 1$, α^r will be a random polynomial of degree less than n. Thus, $\alpha^r \beta^s$ will also be a random polynomial of degree less than n. The chances of such a random polynomial having factors only in T is $P_p(n, m)$ (see §2). Therefore, the expected number of executions of Stage 3 is $(w+1)/P_p(n, m) \le (w+1)p^m n^{n/m} \in L_{p^n}[1/2, 3]$, since $w+1, p^m \in L_{p^n}[1/2, 1]$ and $n^{n/m} \le e^{\log(n)n/(n\log(n)/\log(p))^{1/2}} \le e^{(n\log(n)\log(p))^{1/2}} \in L_{p^n}[1/2, 1]$. Since factorization in $Z/pZ[x]$ can be done in random polynomial time [5], there exists an $L_2 \in L_{p^n}[1/2, 3]$ such that the expected number of steps required for a pass through Stage 3 is at most L_2.

Stage 4: As indicated in §2, there must exist $a_1, a_2, \dots, a_{w+1} \in Z_{\ge 0}^{<p^n-1}$ such that $\mathrm{GCD}(a_1, a_2, \dots, a_{w+1}) = 1$ and $\sum_{i=1}^{w+1} a_i v_i \equiv \langle 0, 0, \dots, 0\rangle \bmod (p^n - 1)$. Further, as follows from §2, there exists an algorithm which will calculate $a_1, a_2, \dots, a_{w+1}$ in $O(w^3 \log^3(p^n))$ steps. Hence, there exists an $L_3 \in L_{p^n}[1/2, 3]$ such that the number of steps required for a single pass through Stage 4 is at most L_3.

Next, it will be argued that the expected number of passes through Algorithm II is negligible. Stages will be repeated only if $(l, p^n - 1) \ne 1$ in Stage 7. However, $(l, p^n - 1) = 1$ with probability $\phi(p^n - 1)/(p^n - 1) \ge 1/c\log(p^n)$, where $c \in \mathfrak{R}_{>0}$ is independent of p and n [3, Lemma 4]. Briefly, as in the analysis of Algorithm I, this can be argued as follows: Since from Stage 3(b), $\mathrm{GCD}(a_1, a_2, \dots, a_{w+1}) = 1$, it follows that for all primes t dividing $p^n - 1$ there exists an $i \in Z_{>0}^{\le w+1}$ such that a_i is relatively prime to t. Consider $\gamma_i \equiv \alpha^{r_i} \beta^{s_i}$, and observe that for all $s \in Z_{\ge 0}^{<p^n-1}$ there exists a unique $r \in Z_{\ge 0}^{<p^n-1}$ such that $\gamma_i \equiv \alpha^r \beta^s$. Hence, s_i is "random" mod t, and consequently $l = \sum_{i=1}^{w+1} s_i a_i$ is also "random" mod t. Hence, the expected number of passes through each stage of the algorithm is at most $c\log(p^n)$.

Thus, there exists an $L_4 \in L_{p^n}[1/2, 3]$ such that the expected number of steps required by Algorithm II on inputs p, f, α, β such that $p \in Z_{>0}$ is prime, $f \in Z/pZ[x]$ is monic, irreducible of degree $n \ge p$, and $\alpha, \beta \in Z/pZ[x]$ of degree less than n with $[\alpha] \in (Z/pZ[x])/(f)$ a generator of the multiplicative group and $\beta \not\equiv 0 \bmod f$ is at most L_4.

Observe that $\alpha^l \beta^k = \prod_{i=1}^{w+1} \gamma_i^{a_i}$ is the product of $s = \prod_{i=1}^{w+1} \tilde{\gamma}_i^{a_i}$ times a $(p^n - 1)$th power. Hence, $\alpha^l \beta^k \equiv s \bmod f$. Next observe that since $[\alpha]$ generates the multiplicative group of $(Z/pZ[x])/(f)$, a $y \in Z_{\ge 0}^{<p-1}$ such that $\alpha^{y((p^n-1)/(p-1))} \equiv s \bmod f$ must exist. Finally, it is clear from Stage 7 that the output of the algorithm is x such that $\alpha^x \equiv \beta \bmod f$.

Acknowledgments

We would like to thank Dennis Estes and Bob Guralnick for their help. In an earlier version of this algorithm the authors had chosen $m = \lfloor n^{1/2} \rfloor$. We would like to thank Don Coppersmith for pointing out that the optimum choice is $m = \lceil (n\log(n)/\log(p))^{1/2} \rceil$. This choice significantly improves the running time of the algorithm.

Discussion. Little effort was made to "optimize" the algorithm presented here. It is possible to improve the running time in several ways. Sparse matrix methods can be used to find some dependencies [28]. Smoothness of norms can be tested using the "elliptic curve methods" [18]. The integer factoring done in various parts can probably be avoided, if necessary, or "$L[1/3]$" methods can be used (e.g., [3, 20]). Also, heuristically, the expected size of q in Algorithm I can be argued to be less than $\tilde{c}n(\log(np))^c$ for some $c, \tilde{c} \in \mathfrak{R}_{>0}$. This will lead to norms of size $p^n(\tilde{c}n(\log(np))^c)^n \in L_{p^n}[1, 2]$. Using $B \in L_{p^n}[1/2, 1]$ and the ideas above, we believe that a running time in $L_{p^n}[1/2, 2]$ is achievable for Algorithm I.

Several alternatives exist for our handling of the case $n \geq p$. Lovorn's algorithm [21], which has a running time in $L_{p^n}[1/2, \sqrt{2}]$, covers this case. Alternatively, Lovorn's improved bound on $N_p(n, m) \leq p^n e^{-(n/m)(\log(n/m)+\log\log(n/m)+O(1))}$ together with sparse matrix techniques could be used to modify Algorithm II and also yield an $L_{p^n}[1/2, \sqrt{2}]$ result. It would also be of interest to adapt Algorithm I to this setting.

Hence, overall it appears discrete logarithms over $\mathrm{GF}(p^n)$ can be computed in $L_{p^n}[1/2, 2]$ expected time.

There appear to be several natural open problems.

- Do there exist a $c \in Z_{>0}$ and an algorithm for discrete logarithms over $\mathrm{GF}(p^n)$ with provable expected running time in $L_x[1/2, c]$?
- Does there exist an algorithm for discrete logarithms over $\mathrm{GF}(p^n)$ with heuristic expected running time in $L_x[1/2, 1]$?
- Does there exist an algorithm for discrete logarithms over $\mathrm{GF}(p^n)$ with provable expected running time in $L_x[1/2, 1]$?
- Do there exist a $c \in Z_{>0}$ and an algorithm for discrete logarithms over $\mathrm{GF}(p^n)$ with heuristic expected running time in $L_x[1/3, c]$?

BIBLIOGRAPHY

1. L. M. Adleman, *A subexponential algorithm for discrete logarithms with applications to cryptography*, Proc. 20th IEEE Found. Comp. Sci. Symp., IEEE Computer Society, Long Beach, CA, 1979, pp. 55–60.

2. ——, *Factoring numbers using singular integers*, Proc. 23rd Annual ACM Symposium on Theory of Computing, Assoc. Comput. Mach., New York, 1991, pp. 64–71.

3. L. M. Adleman and M.-D. A. Huang, *Primality testing and Abelian varieties over finite fields*, Lecture Notes in Math., vol. 1512, Springer-Verlag, Berlin, 1992.

4. L. M. Adleman and H. W. Lenstra, Jr., *Finding irreducible polynomials over finite fields*, Proc. 18th Annual ACM Symposium on Theory of Computing, Assoc. Comput. Mach., New York, 1986, pp. 350–355.

5. E. Berlekamp, *Factoring polynomials over large finite fields*, Math. Comp. **24** (1970), 713–735.

6. E. Bach and J. Shallit, *Factoring with cyclotomic polynomials*, Proc. 26th IEEE Found. Comp. Sci. Symp., IEEE Computer Society, Los Angeles, CA, 1985, pp. 443–450.

7. E. R. Canfield, P. Erdős, and C. Pomerance, *On a problem of Oppenheim concerning "Factorisatio Numerorum"*, J. Number Theory **17** (1983), 1–28.

8. D. Coppersmith, *Fast evaluation of logarithms in fields of characteristic two*, IEEE Trans. Inform. Theory **IT-30** (1984), 587–594.

9. D. Coppersmith, A. M. Odlyzko, and R. Schroeppel, *Discrete logarithms in* $\mathrm{GF}(p)$, Algorithmica, **1** (1986), 1–15.

10. W. Diffe and M. E. Hellman, *New directions in cryptography*, IEEE Trans. Inform. Theory **IT-22** (1976), 644–654.

11. H. M. Edwards, *Fermat's Last Theorem*, Graduate Texts in Math., vol. 50, Springer-Verlag, New York, 1977.

12. T. ElGamal, *A public key cryptosystem and a signature scheme based on discrete logarithms*, IEEE Trans. Inform. Theory **IT-31** (1985), 469–472.

13. ——, *A subexponential-time algorithm for computing discrete logarithms over* $\mathrm{GF}(p^2)$, IEEE Trans. Inform. Theory **IT-31** (1985), 473–481.

14. K. F. Gauss, *Disquisitiones arithmeticae*, translation A. A. Clarke, Yale Univ. Press, New Haven, CT, 1966.

15. D. M. Gordon, *Discrete logarithms in* $\mathrm{GF}(p)$ *using the number field sieve*, manuscript, April 4, 1990.

16. ——, *Discrete logarithms in* $\mathrm{GF}(p^n)$ *using the number field sieve*, preliminary version, manuscript, November 29, 1990.

17. M. E. Hellman and J. M. Reyneri, *Fast computation of discrete logarithms in* $\mathrm{GF}(q)$, Advances in Cryptography: Proceedings of CRYPTO '82 (D. Chaum, R. Rivest, A. Sherman, eds.), Plenum Press, New York, 1983, pp. 3–13.

18. H. W. Lenstra, Jr., *Finding isomorphisms between finite fields*, Math. Comp. **56** (1991), 329–347.

19. ——, *Factoring integers with elliptic curves*, Ann. of Math. (2) **126** (1987), 649–673.

20. A. K. Lenstra, H. W. Lenstra, Jr., M. S. Manasse, and J. M. Pollard, *The number field sieve*, Proc. 22nd STOC, Assoc. Comput. Mach., New York, 1990, pp. 564–572.

21. R. Lovorn, *Rigorous, subexponential algorithms for discrete logarithms over finite fields*, Ph.D. Thesis, University of Georgia, May 1992.

22. M. Newman, *Bounds for class numbers*, Proc. Sympos. Pure Math., vol. 8, Amer. Math. Soc., Providence, RI, 1965, pp. 70–77.

23. A. M. Odlyzko, *Discrete logarithms in finite fields and their cryptographic significance*, Proceedings of Eurocrypt '84 (T. Beth, N. Cot, and I. Ingemarsson, eds.), Lecture Notes in Comput. Sci., vol. 209, Springer-Verlag, Berlin, 1985, pp. 224–314.

24. C. Pomerance, *Fast, rigorous factorization and discrete logarithms*, Discrete Algorithms and Complexity (D. S. Johnson, T. Nishizeki, A. Nozaki, and H. S. Wilf, eds.), Academic Press, Orlando, FL, 1987, pp. 119–144.

25. M. O. Rabin, *Probabilistic algorithms in finite fields*, SIAM J. Comput. **9** (1980), 273–280.

26. R. Solovay and V. Strassen, *A fast Monte-Carlo test for primality*, SIAM J. Comput. **6** (1977), 84–85.

27. L. C. Washington, *Introduction to cyclotomic fields*, Graduate Texts in Math., vol. 83, Springer-Verlag, New York, 1982.

28. D. Wiederman, *Solving sparse linear equations over finite fields*, IEEE Trans. Inform. Theory. **IT-32** (1986), 54–62.

29. A. E. Western and J. C. P. Miller, *Tables of indices and primitive roots*, Royal Society Mathematical Tables, vol. 9, Cambridge Univ. Press, 1968.

Department of Computer Science, University of Southern California, University Park, Los Angeles, California 90089-0781
E-mail address, L. M. Adleman: adleman@pollux.usc.edu
E-mail address, J. DeMarrais: jed@pollux.usc.edu