An Algorithm to Solve the Discrete Logarithm Problem with the Number Field Sieve

An Commeine[1] and Igor Semaev[2]

[1] Katholieke Universiteit Leuven, Departement Wiskunde, Afdeling Algebra, Celestijnenlaan 200B, B-3001 Leuven, Belgium
[2] Universitetet i Bergen,Institutt for informatikk, HIB - Thormhlensgt. 55, N-5020 Bergen, Norway

Abstract. Recently, Shirokauer's algorithm to solve the discrete logarithm problem modulo a prime p has been modified by Matyukhin, yielding an algorithm with running time $L_p[\frac{1}{3}, 1.9018\ldots]$, which is, at the present time, the best known estimate of the complexity of finding discrete logarithms over prime finite fields and which coincides with the best known theoretical running time for factoring integers, obtained by Coppersmith. In this paper, another algorithm to solve the discrete logarithm problem in $\mathbb{F}_p^*$ for p prime is presented. The global running time is again $L_p[\frac{1}{3}, 1.9018\ldots]$, but in contrast with Matyukhins method, this algorithm enables us to calculate individual logarithms in a separate stage in time $L_p[\frac{1}{3}, 3^{1/3}]$, once a $L_p[\frac{1}{3}, 1.9018\ldots]$ time costing pre-computation stage has been executed. We describe the algorithm as derived from [6] and estimate its running time to be $L_p[\frac{1}{3}, (\frac{64}{9})^{1/3}]$, after which individual logarithms can be calculated in time $L_p[\frac{1}{3}, 3^{1/3}]$.

Keywords: Discrete Logarithms, Number Field Sieve

1 Introduction

Given a prime p and integers a and b, the discrete logarithm of b to the base a in the multiplicative group of the finite field $\mathbb{F}_p$ is defined as the smallest nonnegative integer x such that $a^x \equiv b \pmod{p}$, if it exists.

The security of many, widely used public key cryptosystems, as the well-known Diffie-Hellman key exchange algorithm and the ElGamal Digital signature algorithm, depends on the assumption that for suitably chosen primes, discrete logs are hard to compute. As such, one of the most stimulating factors in research on the complexity of discrete logs is the fact that fast discrete logarithm algorithms could easily undermine these cryptosystems ([12],[13] for a survey).

General methods that can also be applied in other groups than $\mathbb{F}_p^*$, are Shanks deterministic "baby steps, giant steps" attack ([14]) and two other randomized algorithms due to Pollard ([16]), such as the Pollard ρ-method. For both methods, the number of operations to compute a discrete logarithm roughly equals $q^{1/2}$, where q is the largest prime factor of $p-1$, but Pollards methods use almost no space in contrast with Shanks method, which has space requirement

$q^{1/2}$. Moreover, the Pollard ρ-method was parallelized in 1993 by van Oorschot and Wiener ([23]) in such a way that the expected number of steps that each processor performs to obtain a discrete logarithm is about $q^{1/2}/t$, where t is the number of processors. These attacks have an exponential worst case complexity, since the largest prime factor of $p-1$ can be almost as large as p.

Making use of additional knowledge of the underlying group, index calculus methods, based on an idea of Kraitchik ([11]), provide subexponential algorithms. These methods typically consist of three phases: generating relations, solving equations and computing individual logarithms using the results of the first two steps. The first two steps, called the pre-computation stage, determine the running time of the algorithm. Once the pre-computation stage is finished for a prime p, individual logarithms modulo that prime can be computed more efficiently. Running time bounds of the earliest index calculus algorithms are of the form $L_p[\frac{1}{2}, c]$ for some constant $c > 0$. Large c however yield impractical algorithms, so many researchers tried to lower this value c during 1970s and 1980s ([11],[14] for references). Both the Linear Sieve Method and the Gaussian Integer Method ([4]), where the use of an imaginary quadratic number field was introduced, achieved the value $c = 1$. In 1998, work on the latter allowed Joux and Lercier to compute discrete logs modulo a 90-digit prime number in [6]. The asymptotic running time bound with $c = 1$ was a record value for a long time.

Speeding up the pre-computation stage was possible due to advances in linear algebra, namely solving sparse systems with n unknowns in not much more than n^2 steps ([15]). This is achieved by the Wiedemann algorithm ([24]), based on the Berlekamp-Massey algorithm and the Cayley-Hamilton theorem and, by adaptations of the finite field version of Lanczos and conjugate gradient algorithms ([4],[14]), that can be combined with structured Gauss Elimination ([14]).

In 1988, Pollard found a new approach for factoring integers. This technique was developed into the special number field sieve by Hendrik Lenstra. It factors integers of special forms in time $L_N[\frac{1}{3}, c]$ with $c = (\frac{32}{9})^{1/3} = 1.5262\ldots$, where N is the number to be factored. Later the method was extended to factor arbitrary integers in time $L_N[\frac{1}{3}, c]$ with $c = (\frac{64}{9})^{1/3} = 1.9229\ldots$ in the general number field sieve, that arose through a collaboration of several researchers ([8] for details). The value of c was improved to $c = 1.9018\ldots$ by Coppersmith in [3].

The general number field sieve was adapted to the computation of discrete logs modulo a prime by Gordon in [5] in 1992. He obtained running time $L_p[\frac{1}{3}, c]$ with $c = 2.0800\ldots$. The value of c was lowered by Shirokauer in [19] to $c = (\frac{64}{9})^{1/3} = 1.9229\ldots$ in 1993. Adapting this algorithm following the ideas of Coppersmith, Matyukhin in [10] achieved the same constant as Coppersmith in [3], thus $c = 1.9018\ldots$. With the latter two algorithms however, it's impossible to efficiently compute individual logarithms, since the linear algebra must be redone for every new logarithm. For special prime numbers, this deficiency was overcome by Semaev in [21], moreover yielding a running time of $L_p[\frac{1}{3}, (\frac{32}{9})^{1/3}]$ and $L_p[\frac{1}{3}, \frac{1+2\sqrt{2}}{18^{1/3}}] = L_p[\frac{1}{3}, 1.4608\ldots]$ for an individual logarithm. Joux and Lercier were able to separate the pre-computation stage and the computation of individual logarithms for primes lacking any special structure in [6], which formed the base of their computation of discrete logs modulo a 130-digits prime, the current record for general primes ([7]). Since their objective was to describe the main ideas behind their C-implementation, they didn't write down the actual algorithm they used to compute individual logarithms nor performed an asymptotic time analysis however.

To achieve a separate individual logarithm stage, we adapt the method in [6] for the pre-computation part and modify the individual logarithm algorithm of [21]. Instead of working with real numbers, we choose to work with a 'logarithmic map' as in [19], though an approach developed in [21] apparently gives the same asymptotic results. The improvements of Coppersmith in [3] are taken into account, to achieve a global running time of $L_p[\frac{1}{3}, 1.9018\ldots]$. In contrast with Matyukhin however, individual logarithms can be calculated separately in time $L_p[\frac{1}{3}, 3^{1/3}] = L_p[\frac{1}{3}, 1.44225\ldots]$ after a $L_p[\frac{1}{3}, 1.9018\ldots]$-time costing pre-computation stage. In order to compare the method in [6] with ours, we give a precise theoretical description of the algorithm as we've understood and built it out of the ideas given in [6]. A running time analysis of this algorithm is performed, using the theoretical settings developed in the analysis of our algorithm. We show that the optimal cost for this algorithm is $L_p[\frac{1}{3}, (\frac{64}{9})^{1/3}]$, with the possibility to calculate individual logarithms separately in time $L_p[\frac{1}{3}, 3^{1/3}]$.

The core idea, which allows us to achieve this running time for the individual logarithm stage, is expressing logarithms of medium-sized prime numbers into logarithms of smaller numbers and the reduction of first degree prime ideals into first degree prime ideals with smaller norm. Inspiration for this was found in [2]. This idea of reducing unknown into known information is also applicable in the one-polynomial variant of the Number Field Sieve, yielding a very similar separate individual logarithm algorithm, again with running time $L_p[\frac{1}{3}, 3^{1/3}]$, not changing the pre-computation time of $L_p[\frac{1}{3}, (\frac{64}{9})^{1/3}]$. (The most expensive reduction will take more time in this setting however; see Remark, Section 4.2.)

We want to remark that running times of all recent algorithms of the form $L_p[\frac{1}{3}, c]$, as the one presented in this paper, are based on heuristic assumptions. There's no proof that they'll run fast. It's possible to obtain rigorous probabilistic algorithms, with running time bounded by $L_p[\frac{1}{2}, c]$ with high probability ([18]).

2 Preliminaries

Definition 1. *An integer n is B-***smooth** *if and only if $q \leq B$ for all (natural) prime numbers q that divide n.*

When assessing a running time analysis of the algorithm, we make use of the complexity-function

$$L_p[t, s] = e^{s(1+o(1))(\ln p)^t (\ln \ln p)^{1-t}} ,$$

where $o(1)$ denotes a function tending to 0 as $p \to \infty$. The expression $o(1)$ in the exponent hides a lot: this notation is meant as a first order approximation to the real computational complexity.

The following theorem gives an estimation of the probability that a number smaller or equal to x is y-smooth in terms of the above complexity function.

Theorem 1. *Let $0 < y_1 < x_1 \leq 1$ and $y_2, x_2 > 0$. Let $x = L_p[x_1, x_2]$ and $y = L_p[y_1, y_2]$, then*

$$\frac{\psi(x,y)}{x} = L_p[x_1 - y_1, -\frac{x_2}{y_2}(x_1 - y_1)] \ ,$$

where $\psi(x, y)$ =the number of natural numbers smaller or equal to x which are y-smooth.

This follows from a more general theorem of Canfield, Erdös and Pomerance:

Theorem 2. *([1]) If $x \geq 10$ and $y > \ln x$, then it holds that*

$$\psi(x,y) = xu^{-u(1+o(1))} \ \text{with} \ u = \frac{\log x}{\log y} \ ,$$

where the limit implicit in the $o(1)$ is for $x \to \infty$.

We recall some useful results from algebraic number theory. Let $f = X^d + f_1X^{d-1} + \cdots + f_d$ be a monic, irreducible polynomial of degree d with root α. We denote the field $\mathbb{Q}(\alpha) = K$ and ϑ_K the ring of algebraic integers of K. Following propositions are useful:

Proposition 1. *([21]) If q does not divide $[\vartheta_K : \mathbb{Z}[\alpha]]$ and*

$$f(X) = \prod_i h_i^{e_i}(X) \ \text{in} \ \mathbb{F}_q[X] \ ,$$

where $h_i(X)$ are distinct irreducible polynomials in $\mathbb{F}_q[X]$, then

$$q\vartheta_K = \prod_i \mathcal{U}_i^{e_i} \ ,$$

for distinct prime ideals $\mathcal{U}_i = h_i(\alpha)\vartheta_K + q\vartheta_K$ in ϑ_K and Norm($\mathcal{U}_i$) $= q^{deg \ h_i(X)}$.

This proposition suggests making a distinction between prime ideals in ϑ_K.

Definition 2. *A prime ideal $\mathcal{P}$ of ϑ_K of degree 1 is* bad *if its norm divides the index $[\vartheta_K : \mathbb{Z}[\alpha]]$. All other prime ideals of degree 1 are called* good.

Good prime ideals appear in factorizations as mentioned below.

Proposition 2. *([21]) If $a, b \neq 0$ are coprime integers such that*

$$b^d f\left(\frac{a}{b}\right) = a^d + f_1ba^{d-1} + \cdots + f_db^d$$

is coprime to $[\vartheta_K : \mathbb{Z}[\alpha]]$, then

$$(a - b\alpha)\vartheta_K = \mathcal{U}_1^{l_1}\mathcal{U}_2^{l_2}\ldots\mathcal{U}_s^{l_s} \ ,$$

where $\mathcal{U}_i$ are distinct good prime ideals of ϑ_K for $i = 1, \ldots, s$ and Norm($\mathcal{U}_i$) $= q_i$ for distinct q_i. Moreover,

$$|b^d f\left(\frac{a}{b}\right)| = \prod_{i=1}^{s} q_i^{l_i} \ .$$

For ease of exposition, suppose $p - 1 = 2q$ with q a large prime that doesn't ramify in K. Let $\Gamma_K = \{\gamma \in \vartheta_K \mid \gcd(\mathrm{Norm}(\gamma), q) = 1\}$. We use a map l as in [19]: set $\epsilon_K = \mathrm{lcm}\left\{\left|(\vartheta_K/\mathcal{Q})^*\right| \mid \mathcal{Q} \text{ prime ideal in } \vartheta_K \text{ lying above } q\right\}$, then

$$\begin{aligned} l : \Gamma_K &\longrightarrow \qquad q\vartheta_K/q^2\vartheta_K \\ \gamma &\longmapsto (\gamma^{\epsilon_K} - 1) + q^2\vartheta_K \ . \end{aligned}$$

Consider $q\vartheta_K/q^2\vartheta_K$ as a $\mathbb{Z}/q\mathbb{Z}$-vectorspace. We generate a sequence of length a little more than the unity rank of ϑ_K of random units $u \in \vartheta_K^*$ and calculate the images $l(u)$. The linear independent vectors amongst these images $l(u)$ span the subspace $l(\vartheta_K^*) \subseteq q\vartheta_K/q^2\vartheta_K$ with high probability. Assume they form a basis $\{qb_j + q^2\vartheta_K \mid j = 1, \ldots, t_K\}$ of $l(\vartheta_K^*)$. Expand this basis to a basis $\{qb_j + q^2\vartheta_K \mid j = 1, \ldots, d\}$ of the whole $\mathbb{Z}/q\mathbb{Z}$-vectorspace $q\vartheta_K/q^2\vartheta_K$. Denote

$$\begin{aligned} \lambda_{K,j} : \Gamma_K &\longrightarrow \ \mathbb{Z}/q\mathbb{Z} \\ \gamma &\longmapsto \lambda_{K,j}(\gamma) \end{aligned}$$

such that $l(\gamma) = \sum_{j=1}^{d} \lambda_{K,j}(\gamma)(qb_j + q^2\vartheta_K)$. Remark that $l(\gamma\gamma') = l(\gamma) + l(\gamma')$, such that $\lambda_{K,j}(\gamma\gamma') = \lambda_{K,j}(\gamma) + \lambda_{K,j}(\gamma')$ for $j = 1, \ldots, d$.

The largest contribution to the time needed for the practical determination of all $\lambda_{K,j}(\gamma)$ for $\gamma \in \Gamma_K$, comes from the exponentiation to the power $\epsilon_K < q^d$ in the ring $\mathbb{Z}[X]/(f, q^2)$, costing $O(d^3 \ln^3 p)$ bit operations.

3 The Algorithm

3.1 Needs and Assumptions

Choose two natural numbers $d = \delta(1+o(1))\,(\ln p/\ln\ln p)^{1/3}$ and $m = p^{(1+o(1))/d}$, both depending on p, where the limit implicit in the $o(1)$ is for $p \to \infty$. The parameter δ will be defined later. Suppose f is an irreducible polynomial of degree d with coefficients bounded by m, such that $f(m) \equiv 0 \bmod p$, obtained as in the Number Field Sieve setting (NFS). Remark that use of polynomials as in [6], namely a degree $d+1$-polynomial with small coefficients and having a root μ modulo p and a degree d-polynomial with the same root μ modulo p, having coefficients of the order $p^{1/(d+1)}$, is thought of giving the best practical results.

For simplicity, we assume $f = f_0$ to be monic. We work with polynomials

$$f_i(X) = f_0(X) + i(X - m) \ \text{ for } i = 1, \ldots, V$$

that are irreducible and such that neither p nor q divide their discriminants. These conditions are easily checked ([5]). For simplicity, we assume all values of i determine valid polynomials. Remark that the coefficients of these polynomials get somewhat larger, becoming $\leq (V+1)m = VL_p[\frac{2}{3}, \frac{1}{\delta}]$ in first order estimate.

Let α_i be a root of f_i, $K_i = \mathbb{Q}(\alpha_i)$ an algebraic number field of degree d over $\mathbb{Q}$ and ϑ_{K_i} the ring of algebraic integers of K_i. Remark that α_i is an algebraic integer in K_i by the assumption that f_i is monic. The number p doesn't divide the discriminant of the polynomial f_i, hence it doesn't divide $[\vartheta_{K_i} : \mathbb{Z}[\alpha_i]]$. According to Proposition 1, $\mathcal{P}_i = (\alpha_i - m)\vartheta_{K_i} + p\vartheta_{K_i}$ then is a first degree prime ideal, and we denote $\pi_i(\varepsilon) = \overline{\varepsilon}$ for π_i the projection-map

$$\pi_i : \vartheta_{K_i} \longrightarrow \frac{\vartheta_{K_i}}{\mathcal{P}_i} \, (\cong \mathbb{F}_p) \ , \ \overline{\alpha_i} = m \ . \tag{1}$$

For every field K_i, we denote the maps $\lambda_{K_i,j}$ and the set Γ_{K_i}, defined as above, as $\lambda_{i,j}$ and Γ_i respectively. Let r_i be the torsion free rank of $\vartheta^*_{K_i}$. Since q doesn't divide the discriminant of f_i, $\vartheta^*_{K_i}$ contains no primitive q'th roots of unity. This implies that the dimension t_{K_i} of the $\mathbb{Z}/q\mathbb{Z}$-subspace $l(\vartheta^*_{K_i}) \subseteq q\vartheta_{K_i}/q^2\vartheta_{K_i}$ is less then or equal to r_i. We assume that $\gcd(h_{K_i}, q) = 1$ and $\{u \in \vartheta^*_{K_i} \mid u \equiv 1 \bmod q^2\} \subseteq (\vartheta^*_{K_i})^q$ for every i. One can check that, under these conditions, the well-defined homomorphisms

$$\begin{array}{rcl} \overline{\lambda}_i : \vartheta^*_{K_i}/(\vartheta^*_{K_i})^q & \longrightarrow & (\mathbb{Z}/q\mathbb{Z})^{r_i} \\ \gamma(\vartheta^*_{K_i})^q & \longmapsto & (\lambda_{i,1}(\gamma), \ldots, \lambda_{i,r_i}(\gamma)) \end{array}$$

are isomorphisms (thus $t_{K_i} = r_i$).

3.2 The Algorithm

Choose bounds $E = L_p[\frac{1}{3}, \epsilon]$, $B_1 = L_p[\frac{1}{3}, \beta]$ and $B_2 = L_p[\frac{1}{3}, \gamma]$, where ϵ, β, γ are parameters with $\beta \geq \gamma$.

Finding Relations

1. Let S_i be the set of good prime ideals in ϑ_{K_i} with norm $\leq B_2$ and coprime to q. As in the modified number field sieve due to Coppersmith, we set $V = \pi(B_1)/(\pi(B_2) + d) = L_p[\frac{1}{3}, \beta - \gamma]$ and determine triples (a, b, i) with $|a| \leq E$, $1 \leq b \leq E$, called good, such that, for q_j ranging over prime numbers $\leq B_1$ and $\mathcal{U}_i$ ranging over prime ideals in S_i, it holds that

$$a - bm = \pm \prod_{q_j \leq B_1} q_j^{e_{abj}} \tag{2}$$

$$(a - b\alpha_i)\vartheta_{K_i} = \prod_{\mathcal{U}_i \in S_i} \mathcal{U}_i^{n_{ab\mathcal{U}_i}} . \tag{3}$$

To achieve about $2(|S_i| + r_i)$ triples per field K_i, we take $\epsilon = (3\gamma^2\delta\beta + \gamma + \beta)/((6\gamma - \delta)\delta\beta)$ and $6\gamma - \delta > 0$. It is shown in [3] that finding appropriate triples takes time

$$L_p[\frac{1}{3}, \max\{\beta, 2\epsilon\}] + L_p[\frac{1}{3}, 2\epsilon - \frac{1}{3\delta\beta} + \beta - \gamma] \ . \qquad (4)$$

2. Since $\overline{\lambda}_i$ are isomorphisms for $i = 0, \ldots, V$, it follows from [20] that there exist unique elements $\mathcal{X}_{\mathcal{U}_i}, \mathcal{X}_{i,j} \in \mathbb{Z}/q\mathbb{Z}$, not depending on the set S_i of ideals, such that for all triples (a, b, i) collected, it holds that

$$\log_g \pi_i(a - b\alpha_i) \equiv \sum_{\mathcal{U}_i \in S_i} \mathcal{X}_{\mathcal{U}_i} n_{ab\mathcal{U}_i} + \sum_{j=1}^{r_i} \mathcal{X}_{i,j} \lambda_{i,j}(a - b\alpha_i) \pmod q \ ,$$

using (3). Together with (2) and taking into account that $\log_g \pm 1 \equiv 0 \pmod q$, this equivalence leads to the equation

$$-\sum_{q_j \le B_1} e_{abj} \log_g q_j + \sum_{\mathcal{U}_i \in S_i} \mathcal{X}_{\mathcal{U}_i} n_{ab\mathcal{U}_i} + \sum_{j=1}^{r_i} \mathcal{X}_{i,j} \lambda_{i,j}(a - b\alpha_i) \equiv 0 \pmod q \ .$$

To establish these equations, we only need to evaluate $\lambda_{i,j}(a - b\alpha_i)$ for $j = 1, \ldots, r_i$ for all good triples (a, b, i). This takes asymptotic time $O(d^3 \ln^3 p) \left(\sum_{i=0}^{V} 2(|S_i| + r_i)\right) \approx O(d^3 \ln^3 p) 2(V+1)(\pi(B_2) + d) = \pi(B_1)$.

Solving the System Through finding relations as above, we get a homogeneous system of about $\sum_{i=0}^{V} 2(|S_i| + r_i) \approx 2(V+1)(\pi(B_2) + d) \approx 2\pi(B_1)$ equations, which has to be solved for $\pi(B_1) + \sum_{i=0}^{V} (|S_i| + r_i) \approx \pi(B_1) + (\pi(B_2) + d)(V+1) \approx 2\pi(B_1)$ unknowns $\log_g q_j$ and $\mathcal{X}_{\mathcal{U}_i}$,$\mathcal{X}_{i,j}$. In order to get a unique non-zero solution to the system, take g a B_1-smooth number $g = \prod_{q_j \le B_1} q_j^{e_{gj}}$, generating $\mathbb{F}_p^*$, what can be done under the assumption of the Extended Riemann Hypothesis ([22]), and expand the system with the equation

$$\sum_{q_j \le B_1} e_{gj} \log_g q_j \equiv \log_g g \equiv 1 \pmod q.$$

Let U be the matrix with blocks $U_i = (e_{abij})_{(a,b,i),j}$ on its rows, where $e_{abij} = e_{abj}$ in (2) for a good triple (a, b, i) and let P, respectively L, be matrices with blocks $P_i = (n_{ab\mathcal{U}_i})_{(a,b,i),\mathcal{U}_i}$, respectively $L_i = (\lambda_{i,j}(a - b\alpha_i))_{(a,b,i),j}$, on the diagonal for i from 0 to V. The rows of these matrices run over good triples (a, b, i). Let U_g be the rowvector $(e_{gj})_j$, then the matrix of the system has layout:

$$\begin{pmatrix} 1 \ , \ -U_g \ , \ 0 \ , \ 0 \\ 0 \ , \ -U \ \ , \ P \ , \ L \end{pmatrix} = \left(\begin{array}{c|c|cccc|cccc} 1 & -U_g & 0 & 0 & \ldots & 0 & 0 & 0 & \ldots & 0 \\ 0 & -U_0 & P_0 & 0 & \ldots & 0 & L_0 & 0 & \ldots & 0 \\ 0 & -U_1 & 0 & P_1 & \ldots & 0 & 0 & L_1 & \ldots & 0 \\ \vdots & \vdots & & & \ddots & & & & \ddots & \\ 0 & -U_V & 0 & 0 & \ldots & P_V & 0 & 0 & \ldots & L_V \end{array}\right) .$$

This sparse system can be solved combining structured Gaussian elimination with a sparse matrix technique, such as Wiedemann's algorithm ([24]) or Lanczos and conjugate gradient methods ([4],[14]). According to [15], asymptotical time cost to solve the system is

$$O(\pi(B_1)^2) = L_p[\frac{1}{3}, 2\beta] \ . \tag{5}$$

As stated in [20], we can choose whatever 'logarithmic' maps $\mu_{i,j}$ instead of the mappings $\lambda_{i,j}$ used here (as in [19], see above). In this way we can make the system more sparse, so sparse matrix techniques to solve the system work faster. We've for example found maps $\mu_{i,j}$ such that each L_i contained at most $r_i(|S_i| + 1)$ non-zero entries. However, one has to make sure that the advantage of having a sparser system doesn't get lost by the cost of evaluating the mappings $\mu_{i,j}$. This still has to be examined.

3.3 Running Time Analysis Pre-Computation

With running time considerations (4),(5), and taking $\gamma \leq \beta$, ϵ as above and $6\gamma - \delta > 0$, total pre-computation time becomes

$$L_p[\frac{1}{3}, \max\{2\epsilon, 2\epsilon - \frac{1}{3\delta\beta} + \beta - \gamma, 2\beta\}] \ ,$$

which has optimal value $L_p[\frac{1}{3}, 2\beta] = L_p[\frac{1}{3}, 1.9018\ldots]$ as in [3], by taking

$$\beta = \left(\frac{46 + 13\sqrt{13}}{108}\right)^{\frac{1}{3}} \quad , \quad \gamma = \beta\left(\frac{\sqrt{13} - 1}{3}\right) \quad , \quad \delta = \beta\left(\frac{4\sqrt{13} - 10}{3}\right) \ .$$

4 The Individual Logarithm

4.1 The Algorithm

In this section we determine $\log_a b$ (mod $p - 1$) for a generator a of $\mathbb{F}_p^*$ by making use of the $\log_g q_k$, $\mathcal{X}_{\mathcal{U}_i}$ and $\mathcal{X}_{i,j}$ calculated in the former section.

Use the procedure below to calculate $\log_g z \pmod{p-1}$ for $z = a$ and $z = b$. Once these logarithms are calculated, the asked for $\log_a b$ is found as $\log_a b \equiv \log_g b / \log_g a \pmod{p-1}$.

1. Let $Q \leq B_1$ be the largest prime number in the factorbase for which the logarithm is known. Factor $Q^h z$ using the Elliptic Curve Method (ECM) ([9]) for random integers $h \in \{1, \ldots, p-1\}$, until you find one for which $Q^h z \bmod p$ is $L_p[\frac{2}{3}, (\frac{1}{3})^{1/3}]$-smooth. Thus

$$Q^h z \equiv q_1^{n_1} \ldots q_r^{n_r} \pmod{p} \ , \ q_i \text{ prime numbers} \leq L_p[\tfrac{2}{3}, (\tfrac{1}{3})^{\frac{1}{3}}] \ . \tag{6}$$

To check for factors $\le L_p[\frac{2}{3},(\frac{1}{3})^{1/3}]$, each application of ECM takes asymptotic time $L_p[\frac{1}{3},2(\frac{1}{3})^{2/3}]$ ([10]), such that the total time to find a good h is

$$L_p[\frac{1}{3},\left(\frac{1}{3}\right)^{\frac{2}{3}}]L_p[\frac{1}{3},2\left(\frac{1}{3}\right)^{\frac{2}{3}}]=L_p[\frac{1}{3},3^{\frac{1}{3}}]=L_p[\frac{1}{3},1.44225\ldots]\ ,$$

where we estimate the probability for a number $< p$ to be $L_p[\frac{2}{3},(\frac{1}{3})^{1/3}]$-smooth as $L_p[\frac{1}{3},-(\frac{1}{3})^{2/3}]$, using Theorem 1.

2. For all $q_i(>B_1)$ in (6), we need to find $\log_g q_i$. This is done by expressing these logarithms in terms of known logarithms by means of reductions, which are described in the next subsection.
3. Calculate $\log_g z \equiv -h\log_g Q+\sum_{i=1}^{r} n_i \log_g q_i \pmod q$ as a sum of known logarithms. Then, compute $\log_g z \pmod{p-1}$ as $(\log_g z \mod q)+\phi q$, testing whether $\phi=0$ or $\phi=1$ using modular exponentiation.
The computation $\log_a b\equiv \log_g b/\log_g a \pmod{p-1}$ after applying the procedure to $z=a,b$, together with the above calculations,take time $O(\ln^3 p)$.

4.2 Reductions

We explain how to reduce a number and a prime ideal. Time for whatever reduction is of the form $L_p[\frac{1}{3},c]$, with $c\le 3^{1/3}$ for a good choice of parameters.

Reduction of a Number l' We need to reduce numbers l' with $B_1<l'\le L_p[\frac{2}{3},\left(\frac{1}{3}\right)^{1/3}]$. Depending on the largeness of the number that needs to be reduced, we use different parameters. Let $M=L_p[\frac{1}{2},c_M]$ for some constant c_M. If $l'\in[B_1,M]$, we use a parameter ν_1 with $\delta/(6\beta)=0.2456\ldots<\nu_1<1$ and set $e_1=(\frac{3\nu_1\beta}{6\nu_1\beta-\delta})(\frac{2}{3\nu_1\delta\beta}+\frac{\delta}{6\nu_1}-\beta+\gamma)$; for larger l' we use a parameter ν_2 with $0<\nu_2<1$ and set $e_2=(\gamma-\beta)/2+\delta/(12\nu_2)$.

Choose a pair of coprime integers (a,b) with $|a|,|b|\le L_p[\frac{1}{3},e_i]l'^{1/2}$ in the lattice generated by $(m,1)$ and $(l',0)$, which implies that l' divides $a-bm$. We expect about $L_p[\frac{1}{3},2e_i]$ such couples. If $|a-bm/l'|$ is l'^{ν_i}-smooth, check whether $|\mathrm{Norm}(a-b\alpha_j)|=|b^d f_j(a/b)|$ is l'^{ν_i}-smooth, for j such that $\mathrm{Norm}(a-b\alpha_j)$ is simultaneously coprime with q and $[\vartheta_{K_j}:\mathbb{Z}[\alpha_j]]$. If so, Proposition 2 implies that we have a couple (a,b) and j such that at the same time

$$a-bm=l'\prod_l l^{e_{l',l}} \qquad l\le l'^{\nu_i},\ \text{prime} \tag{7}$$

$$(a-b\alpha_j)\vartheta_{K_j}=\prod_{\mathcal{U}_j}\mathcal{U}_j^{m_{l',\mathcal{U}_j}} \quad \mathrm{Norm}(\mathcal{U}_j)\le l'^{\nu_i},\ \mathcal{U}_j\ \text{good prime ideal}\ . \tag{8}$$

This allows us to express $\log_g l'$ in terms of $\log_g l$ with $l\le l'^{\nu_i}$ and $\mathcal{X}_{\mathcal{U}_j}$ for good prime ideals $\mathcal{U}_j$ with $\mathrm{Norm}(\mathcal{U}_j)\le l'^{\nu_i}$ as follows. Equality (8) implies that

$$\log_g \pi_j(a-b\alpha_j)\equiv\sum_{\mathcal{U}_j}\mathcal{X}_{\mathcal{U}_j}m_{l',\mathcal{U}_j}+\sum_{k=1}^{r_j}\mathcal{X}_{j,k}\lambda_{j,k}(a-b\alpha_j) \pmod q\ ,$$

where $\mathcal{U}_j$ runs over ideals as in (8). Combining this equivalence with (7) yields

$$\log_g l' \equiv \sum_{\mathcal{U}_j} \mathcal{X}_{\mathcal{U}_j} m_{l',\mathcal{U}_j} + \sum_{k=1}^{r_j} \mathcal{X}_{j,k}\lambda_{j,k}(a-b\alpha_j) - \sum_{l\leq l'^{\nu_i}} e_{l',l}\log_g l \pmod q \ , \quad (9)$$

where l runs over prime numbers as in (7) and $\mathcal{U}_j$ are prime ideals as in (8).

Using Theorem 2, one can check that the probability for the number $|(a-bm)/l'|$, respectively $|b^d f_j(a/b)|$, to be l'^{ν_i}-smooth can be estimated to be at least $\mathcal{P}_{11} = L_p[\frac{1}{3}, -\frac{1}{3\delta\nu_1\beta}]$, respectively $\mathcal{P}_{21} = L_p[\frac{1}{3}, -(\frac{1}{3\nu_1\delta\beta} + \frac{e_1\delta}{3\nu_1\beta} + \frac{\delta}{6\nu_1})]$ for $l' \in [B_1, M]$ and at least $\mathcal{P}_{12} = L_p[\frac{1}{6}, -\frac{1}{6\delta\nu_2 c_M}]$, respectively $\mathcal{P}_{22} = L_p[\frac{1}{3}, \frac{-\delta}{6\nu_2}]$ for larger l'. Remark that $L_p[\frac{1}{3}, 2e_i]\mathcal{P}_{1i}V \geq 1/\mathcal{P}_{2i}$ for $i = 1, 2$, so enough pairs (a, b) are considered to finish the procedure with a successful triple (a, b, j).

To find a good triple (a, b, j), we have to test $L_p[\frac{1}{3}, 2e_i]$ values $|(a-bm)/l'|$ and $1/\mathcal{P}_{2i}$ values $|b^d f_j(a/b)|$ for l'^{ν_i}-smoothness, using ECM. According to [10], this takes time at most $L_p[\frac{1}{4}, \sqrt{\nu_1 c_M}]$ for a number $l' \in [B_1, M]$, while for larger l' it costs time $L_p[\frac{1}{3}, 2\sqrt{\nu_2}(\frac{1}{3})^{2/3}]$. Using the fact that $1/(3\nu_1\delta\beta) - \beta + \gamma > 0$ since $1/(3\delta\beta(\beta-\gamma)) = 2$, reducing a number $l' \in [B_1, M]$ takes time at most

$$L_p[\frac{1}{3}, 2e_1] + L_p[\frac{1}{3}, \frac{1}{3\nu_1\delta\beta} + \frac{e_1\delta}{3\nu_1\beta} + \frac{\delta}{6\nu_1}] = L_p[\frac{1}{3}, 2e_1] \ .$$

For a choice $0.6942\ldots = \frac{4+\delta^2\beta+3^{1/3}\delta^2}{6\delta\beta(\beta-\gamma+3^{1/3})} \leq \nu_1 < 1$, this won't exceed $L_p[\frac{1}{3}, 3^{1/3}]$. For larger numbers l' time cost will be at most

$$L_p[\frac{1}{3}, 2e_2 + 2\sqrt{\nu_2}\left(\frac{1}{3}\right)^{\frac{2}{3}}] + L_p[\frac{1}{3}, \frac{\delta}{6\nu_2} + 2\sqrt{\nu_2}\left(\frac{1}{3}\right)^{\frac{2}{3}}] = L_p[\frac{1}{3}, \frac{\delta}{6\nu_2} + 2\sqrt{\nu_2}\left(\frac{1}{3}\right)^{\frac{2}{3}}] \ ,$$

which has minimal value $L_p[\frac{1}{3}, 1.1338\ldots]$ for a choice $\nu_2 = \left(\delta^2/(3^{\frac{2}{3}}4)\right)^{1/3} < 1$.

Remark that for a choice $(1 >)\nu_1 \geq \frac{4+\delta^2\beta+x\delta^2}{6\delta\beta(\beta-\gamma+x)} = 0.7406\ldots$ with $x = 1.1338\ldots$, reducing a number $l' \in [B_1, M]$ takes time $\leq L_p[\frac{1}{3}, 1.1338\ldots]$.

Reduction of a Prime Ideal in the Ring ϑ_{K_j} In expression (9), there can appear $\mathcal{X}_{\mathcal{U}'_j}$ with $B_2 <$Norm$(\mathcal{U}'_j) = k' \leq L_p[\frac{2}{3}, \nu_2/3^{1/3}]$. To determine such an unknown number, we reduce the ideal $\mathcal{U}'_j$, which is, according to Proposition 1, generated by $\alpha_j - \alpha_{j,k'}$ and k', for $0 \leq \alpha_{j,k'} < k'$ a root of $f_j(X) \equiv 0 \pmod{k'}$.

As with reducing numbers, we distinguish between $k' \in [B_2, M]$ and larger k', with M as in the reduction of numbers. Likewise we introduce parameters $\tilde{\nu}_1$ with $0.28287\ldots = \delta/(6\gamma) < \tilde{\nu}_1 < 1$ and set $\tilde{e}_1 = \left(\frac{3\gamma\tilde{\nu}_1}{6\gamma\tilde{\nu}_1-\delta}\right)\left(\frac{2}{3\gamma\tilde{\nu}_1\delta} + \frac{\delta}{6\tilde{\nu}_1}\right)$, and $\tilde{\nu}_2$ with $0 < \tilde{\nu}_2 < 1$ and set $\tilde{e}_2 = \delta/(6\tilde{\nu}_2)$.

Choose a pair of coprime integers (a, b) with $|a|, |b| \leq L_p[\frac{1}{3}, \tilde{e}_i]k'^{1/2}$, subject to the usual restriction that $|b^d f_j(a/b)|$ is simultaneously coprime with q and $[\vartheta_{K_j} : \mathbb{Z}[\alpha_j]]$ and the new restriction that $\mathcal{U}'_j$ divides $(a-b\alpha_j)\vartheta_{K_j}$, by taking couples in the lattice spanned by $(\alpha_{j,k'}, 1)$ and $(k', 0)$. When both $|b^d f_j(a/b)|/k'$

and $|a-bm|$ are $k'^{\tilde{\nu}_i}$-smooth, which can be checked using ECM, we have a couple (a,b) such that simultaneously

$$a-bm = \prod_l l^{e_{\mathcal{U}'_j,l}} \qquad l \le k'^{\tilde{\nu}_i} \text{ prime numbers,} \tag{10}$$

$$(a-b\alpha_j)\vartheta_{K_j} = \mathcal{U}'_j \prod_{\mathcal{U}_j} \mathcal{U}_j^{m_{\mathcal{U}'_j,\mathcal{U}_j}} \quad \mathrm{Norm}(\mathcal{U}_j) \le k'^{\tilde{\nu}_i}, \mathcal{U}_j \text{ good prime ideals .} \tag{11}$$

Similarly as before, equality (11) implies that

$$\log_g \pi_j(a-b\alpha_j) \equiv \mathcal{X}_{\mathcal{U}'_j} + \sum_{\mathcal{U}_j} \mathcal{X}_{\mathcal{U}_j} m_{\mathcal{U}'_j,\mathcal{U}_j} + \sum_{k=1}^{r_j} \mathcal{X}_{j,k}\lambda_{j,k}(a-b\alpha_j) \pmod q ,$$

where $\mathcal{U}_j$ runs over ideals as in (11). Combining this with (10) yields

$$\mathcal{X}_{\mathcal{U}'_j} \equiv \sum_l e_{\mathcal{U}'_j,l} \log_g l - \sum_{\mathcal{U}_j} \mathcal{X}_{\mathcal{U}_j} m_{\mathcal{U}'_j,\mathcal{U}_j} - \sum_{k=1}^{r_j} \mathcal{X}_{j,k}\lambda_{j,k}(a-b\alpha_j) \pmod q ,$$

with l prime numbers as in (10) and $\mathcal{U}_j$ prime ideals as in (11).

Deduced as with the reduction of numbers, time-cost of a reduction for ideals with norm $k' \in [B_2, M]$ is $L_p[\frac{1}{3}, \frac{2}{3\gamma\tilde{\nu}_1\delta} + \frac{\tilde{e}_1\delta}{3\gamma\tilde{\nu}_1} + \frac{\delta}{6\tilde{\nu}_1}]$, which doesn't exceed $L_p[\frac{1}{3}, 3^{1/3}]$ for a choice $0.9308\ldots = \frac{4+\delta^2\gamma+3^{1/3}\delta^2}{6\delta\gamma 3^{1/3}} \le \tilde{\nu}_1 < 1$. For ideals with larger norm the reduction takes time $L_p[\frac{1}{3}, \frac{\delta}{6\tilde{\nu}_2} + 2\sqrt{\nu_2\tilde{\nu}_2}\left(\frac{1}{3}\right)^{2/3}]$, which is minimal for $\tilde{\nu}_2 = \left(\delta^2/(12\nu_2 b_2)\right)^{1/3} < 1$, and time-cost is then equal to $L_p[\frac{1}{3}, 0.9658\ldots]$.

Remark that for a choice $(1 >)\tilde{\nu}_1 \ge \frac{4+\delta^2\gamma+x\delta^2}{6\delta\gamma x} = 0.9967\ldots$ with $x = 1.1338\ldots$, time for the reduction of an ideal with norm $k' \in [B_2, M]$ will be $\le L_p[\frac{1}{3}, 1.1338\ldots]$.

Remark This strategy of 'reducing' can also be used with the classical Number Field Sieve setting, where only one polynomial is used at the algebraic side. In a similar way as above, one can show that the reduction of a number l or a prime ideal $\mathcal{U}$ with $\mathrm{Norm}(\mathcal{U}) = l$ takes time $L_p[\frac{1}{3}, (\frac{3}{2})^{1/3}] = L_p[\frac{1}{3}, 1.1447\ldots]$ if $L_p[\frac{1}{2}, c_m] \le l < L_p[\frac{2}{3}, (\frac{1}{3})^{1/3}]$ by taking $\nu = (1/2)^{2/3}$. Since for smaller medium-sized l time needed for a reduction can be made less than $L_p[\frac{1}{3}, (\frac{3}{2})^{1/3}]$ by taking $(1 >)\nu \ge (2^{1/3}6 + 6^{1/3}8 + 24^{1/3}3)/36$, this is the most time consuming reduction. We've shown above that the most time-consuming reduction in our many polynomial case has time cost $L_p[\frac{1}{3}, 1.1338\ldots]$. Hence, the most expensive reduction in the one polynomial variant takes more time than the most expensive reduction in our case. The algorithm to separately compute individual logarithms after the pre-computation is done with the original Number Field Sieve setting, using the idea of reductions, is the same as the one above and has the same running time, namely $L_p[\frac{1}{3}, 3^{1/3}]$. Thus, asymptotically there is no difference in time-usage between the one or more polynomial setting to calculate individual logarithms once the pre-computation has been executed (recall however that the pre-computation is more expensive with the one polynomial setting!).

Reductions: an example Suppose we want to find discrete logarithms in $\mathbb{F}_{83}^*$ to the base $g = 2$. Take $d = 2$ and $m = 30$. Set $f(X) = X^2 + 13$, since for this irreducible polynomial, we have $f(30) \equiv 0 \pmod{83}$ and neither $p = 83$ nor $q = 41$ divide the discriminant -52 of f. Hence, we work in the extension field $\mathbb{Q}(\sqrt{-13})$, for which it is known that $\vartheta = \vartheta_{\mathbb{Q}(\sqrt{-13})} = \mathbb{Z} + \sqrt{-13}\mathbb{Z}$, such that $[\vartheta, \mathbb{Z}[\sqrt{-13}]] = 1$. The unity rank of ϑ is 0, such that no maps λ_j are needed. Note that in fact $\vartheta^* = \{-1, 1\}$, such that it holds that $\{u \in \vartheta^* \mid u \equiv 1 \bmod 41^2\} \subseteq (\vartheta^*)^{41}$. Further on, we have $h_{\mathbb{Q}(\sqrt{-13})} = 2$, thus $h_{\mathbb{Q}(\sqrt{-13})}$ is co-prime with 41.

Let $\bar{t} = t + p\mathbb{Z} \in \mathbb{F}_p$ for every $t \in \mathbb{Z}$. Denote with $\mathcal{U}_{l,r}$ the degree one prime ideal generated by the prime number l and $-r + \sqrt{-13}$ for $r \in \mathbb{N}$. We take smoothness-bound $B_1 = 19$ at the rational side, and smoothness-bound $B_2 = 17$ at the algebraic side. Let S be the set of all good degree one prime ideals with norm ≤ 17. Suppose the pre-computation stage is executed.

Suppose we have to calculate $\log_g \overline{71}$. We use a reduction of the number 71. Take $\nu = 0.91$. For the coprime integers $a = 1, b = -26$, we have that

$$(1 + 26 \times 30)/71 = 11 \quad \text{and} \quad \mathrm{Norm}(1 + 26\sqrt{-13}) = 1 + 13 \times 26^2 = 11 \times 17 \times 47$$

are simultaneously $71^{0.91}$-smooth. The conditions for $\mathrm{Norm}(1 + 26\sqrt{-13})$ to be coprime with 41 and $[\vartheta, \mathbb{Z}[\sqrt{-13}]]$ are fulfilled, so Proposition 2 implies that

$$\begin{aligned} 1 + 26 \times 30 &= 71 \times 11, \\ (1 + 26\sqrt{-13})\vartheta &= \mathcal{U}_{11,8}\mathcal{U}_{17,15}\mathcal{U}_{47,9}, \end{aligned}$$

simultaneously. This leads to the result that

$$\log_g \overline{71} \equiv \mathcal{X}_{\mathcal{U}_{11,8}} + \mathcal{X}_{\mathcal{U}_{17,15}} + \mathcal{X}_{\mathcal{U}_{47,9}} - \log_g \overline{11} \pmod{41} . \tag{12}$$

In this expression for $\log_g \overline{71}$, $\mathcal{X}_{\mathcal{U}_{47,9}}$ is (the only) unknown.

Let $\nu' = 0.8$. Applying the Gaussian Algorithm, we find a short vector $(2, -5)$ in the lattice spanned by $(9, 1)$ and $(47, 0)$, for which we know $\mathcal{U}_{47,9}$ divides $(a - b\sqrt{-13})\vartheta$ for elements (a, b). Since $\mathrm{Norm}(2 + 5\sqrt{-13})$ is coprime with 41 and $[\vartheta, \mathbb{Z}[\sqrt{-13}]]$ and since $\mathrm{Norm}(2 + 5\sqrt{-13})/47 = (2^2 + 13 \times 5^2)/47 = 7$ and $2 + 5 \times 30 = 2^3 \times 19$ are both $47^{0.8}$-smooth, we use $(2, -5)$ to reduce $\mathcal{U}_{47,9}$. Proposition 2 implies that simultaneously

$$\begin{aligned} 2 + 5 \times 30 &= 2^3 \times 19 \ , \\ (2 + 5\sqrt{-13})\vartheta_{\mathbb{Q}(\sqrt{13})} &= \mathcal{U}_{7,1}\mathcal{U}_{47,9}, \ , \end{aligned}$$

what results in the expression

$$\begin{aligned} \mathcal{X}_{\mathcal{U}_{47,9}} &\equiv 3 \log_g \overline{2} + \log_g \overline{19} - \mathcal{X}_{\mathcal{U}_{7,1}} \\ &\equiv 3 + 6 - 32 \equiv 18 \pmod{41} \ , \end{aligned}$$

where $\mathcal{X}_{\mathcal{U}_{7,1}} \equiv 32 \pmod{41}$ and $\log_g \overline{19} \equiv 6 \pmod{41}$ were pre-computed.

Getting back to computation (12) of $\log_g \overline{71}$, we see that

$$\log_g \overline{71} \equiv 34 + 5 + 18 - 24 \equiv 33 \pmod{41} \ ,$$

where $\mathcal{X}_{\mathcal{U}_{11,8}} \equiv 34$, $\mathcal{X}_{\mathcal{U}_{17,15}} \equiv 5$, $\log_g \overline{11} \equiv 24 \pmod{41}$ were pre-computed. One can check that indeed $2^{33} \equiv 71 \pmod{83}$. Remark that the above expression for $\log_g \overline{71}$ is exactly expression (9) for this particular case.

4.3 Running Time Analysis Individual Logarithm

We analyze the time needed to perform step 2 of the algorithm. Set $\nu = \max\{\nu_1, \nu_2, \tilde{\nu}_1, \tilde{\nu}_2\}$. When a number or a prime ideal is reduced, (7) or respectively (10) introduces $O((\ln p/\ln\ln p)^{1/3})$ new medium-sized prime numbers $B_1 \leq l < L_p[\frac{2}{3}, \left(\frac{1}{3}\right)^{1/3}]$ with unknown logarithms. Via (8) or (11), any reduction will also invoke $O((\ln p/\ln\ln p)^{2/3})$ new medium-sized prime ideals $\mathcal{U}_j$ (ideals for which $B_2 \leq \mathrm{Norm}(\mathcal{U}_j) < L_p[\frac{2}{3}, \left(\frac{1}{3}\right)^{1/3}]$) for which $\mathcal{X}_{\mathcal{U}_j}$ is unknown. Let Z be the maximal number of the total of new unknowns induced by one reduction, thus $Z = O((\ln p/\ln\ln p)^{2/3})$. To calculate $\log_g q_i$ for q_i as in (6), $1 + Z + Z^2 + \ldots + Z^{\tilde{w}-1} \leq Z^{\tilde{w}}$ reduction-steps will be needed to get all $\log_g l$ and $\mathcal{X}_{\mathcal{U}_j}$ in the original factorbase, where $\tilde{w}$ is a natural number such that $q_i^{\nu^{\tilde{w}}} \leq B_2$. Since $q_i \leq L_p[\frac{2}{3}, \left(\frac{1}{3}\right)^{1/3}]$, it suffices to find $\tilde{w}$ such that $L_p[\frac{2}{3}, \left(\frac{1}{3}\right)^{1/3}]^{\nu^{\tilde{w}}} \leq B_2$ or, in other words, such that $\nu^{\tilde{w}} \ln L_p[\frac{2}{3}, \left(\frac{1}{3}\right)^{1/3}] \leq \ln B_2$. Since this holds for $\tilde{w} \geq \frac{1}{\ln \nu} \ln \frac{\ln B_2}{\ln L_p[\frac{2}{3}, \left(\frac{1}{3}\right)^{1/3}]} = O(\ln\ln p)$, we can take $\tilde{w} = O(\ln\ln p)$. Hence, the number of reductions won't exceed

$$O((\ln p/\ln\ln p)^{2/3})^{O(\ln\ln p)} = e^{O((\ln\ln p)^2)} .$$

Combining all results of the reductions into the value $\log_g q_i \pmod{q}$ uses time $O((\ln p)^3)e^{O((\ln\ln p)^2)} \approx e^{O((\ln\ln p)^2)}$.

Let c be the constant such that time cost for the most expensive reduction is $L_p[\frac{1}{3}, c]$. It takes time at most

$$L_p[\frac{1}{3}, c]e^{O((\ln\ln p)^2)} + e^{O((\ln\ln p)^2)} = L_p[\frac{1}{3}, c]$$

to compute $\log_g q_i$ for a medium-sized number q_i, so all desired unknown logarithms in (6) can be determined in time $O((\ln p/\ln\ln p)^{\frac{2}{3}})L_p[\frac{1}{3}, c] = L_p[\frac{1}{3}, c]$.

We conclude that the total running time for the individual logarithm algorithm is $L_p[\frac{1}{3}, \max\{3^{1/3}, c\}]$. By choosing parameters as described above, c can be taken not to exceed $3^{1/3}$. Hence, given the results of the pre-computation stage, a calculation of an individual logarithm takes time $L_p[\frac{1}{3}, 3^{1/3}] = L_p[\frac{1}{3}, 1.44225\ldots]$.

5 The Algorithm of Joux and Lercier

To make a running time analysis of the method in [6], we describe the algorithm as we understood it, using the theoretical background we developed before, introducing constants $s_d, s_\alpha, s_\beta, s_l, s_k, c_d, c_\alpha, c_\beta, c_l, c_k \in \mathbb{R}$, which we determine to get a minimal running time. Assume that the optimal degree d behaves as $d = c_d(1+o(1))(\ln p/\ln\ln p)^{s_d}$.

Choose d such that $d+1$ is a prime number. Let f_β be an irreducible polynomial of degree $d+1$ with root μ in $\mathbb{F}_p$ and coefficients of order $O(1)$, such that its Galois group has order $d+1$. Take f_α an irreducible polynomial of degree d such that $f_\alpha(\mu) \equiv 0 \pmod{p}$. By construction, the coefficients of this polynomial are of order $p^{1/(d+1)} = L_p[1-s_d, 1/c_d]$. In general, f_α isn't monic. For ease of exposition however, we assume f_α and f_β to be monic. Let α and β be roots of $f_\alpha,\ f_\beta$ respectively. The ring of algebraic integers in $\mathbb{Q}(\alpha)$, respectively $\mathbb{Q}(\beta)$, is denoted as ϑ_α, respectively ϑ_β. Let r_α, respectively r_β, be the torsion-free rank of ϑ_α^*, respectively ϑ_β^*. At the side of f_α, respectively f_β, we work with smoothness-bound $B_\alpha = L_p[s_\alpha, c_\alpha]$, respectively $B_\beta = L_p[s_\beta, c_\beta]$. Let S_α, respectively S_β, denote the set of degree one prime ideals in ϑ_α, respectively ϑ_β, with norm less then B_α, respectively B_β. Denote $\lambda_{\mathbb{Q}(\alpha),j} = \lambda_j$. Let g denote a generator of $\mathbb{F}_p^*$.

Let $L = L_p[s_l, c_l]$. Sieving coprime pairs (a,b) with $|a| \le L$, $1 \le b \le L$, appropriate for the algorithm in [6], takes asymptotic time ([10],[19])

$$L_p[s_\alpha, c_\alpha] + L_p[s_\beta, c_\beta] + L_p[s_l, 2c_l] \quad ,$$

and results in pairs (a,b) such that simultaneously

$$(a+b\alpha)\vartheta_\alpha = \prod_{P\in S_\alpha} P^{e_{(a,b),P}} \quad , \tag{13}$$

$$(a+b\beta)\vartheta_\beta = \prod_{Q\in S_\beta} Q^{e_{(a,b),Q}} \quad . \tag{14}$$

Since, using Theorem 1, the probability for $|\mathrm{Norm}(a-b\beta)|$ to be B_β-smooth, for $|\mathrm{Norm}(a-b\alpha)|$ to be B_α-smooth respectively, is estimated as $L_p[s_l+s_d-s_\beta, -(s_l+s_d-s_\beta)c_d c_l/c_\beta]$ and as $L_p[s_1-s_\alpha, -(s_1-s_\alpha)c_1/c_\alpha]$ respectively, where $s_1 = \max\{1-s_d, s_l+s_d\}$ and $c_1 = 1/c_d, c_d c_l + 1/c_d$ or $c_d c_l$ if respectively $s_l <, =$ or $> 1-2s_d$, the condition to have $|S_\alpha| + |S_\beta| + r_\alpha + r_\beta + O(1)$ surviving pairs, becomes the following on the parameters s:

$$s_l \ge s_\alpha\ ,\ s_l \ge s_\beta\ ,\ s_l \ge s_l + s_d - s_\beta\ ,\ s_l \ge 1 - s_d - s_\alpha\ ,\ s_l \ge s_l + s_d - s_\alpha\ . \tag{15}$$

Once these parameters are determined, we get conditions on the constants c.

Assume conditions as in [20] are fulfilled. Let $\mathcal{X}_P$, $\mathcal{X}_j$ be the so called virtual logarithms. According to [20] and using (13), every couple (a,b) invokes an immediate congruence

$$\log_g(a+b\mu) \equiv \sum_{P\in S_\alpha} e_{(a,b),P}\mathcal{X}_P + \sum_{j=1}^{r_\alpha} \lambda_j(a+b\alpha)\mathcal{X}_j \pmod{q}\ . \tag{16}$$

Since the polynomial f_β has very small coefficients, it is assumed that the resulting number field has a simple structure, namely that the class field number is 1, and that all fundamental units of ϑ_β can be computed. A similar approach as in [17] can then be used. (Note however that if this approach would run too slowly, one can continue as on the f_α-side, as shown in [20].) For every Q in S_β, let $Q = \gamma_Q \vartheta_\beta$ with $\gamma_Q \in \vartheta_\beta$ and U the set of fundamental units in ϑ_β. Expression (14) leads to

$$\log_g(a + b\mu) \equiv \sum_{u \in U} e_{(a,b),u} \log_g \overline{u} + \sum_{Q \in S_\beta} e_{(a,b),Q} \log_g \overline{\gamma_Q} \pmod{q} \; . \qquad (17)$$

Combining (16) and (17) now yields $|S_\alpha| + |S_\beta| + r_\alpha + r_\beta + O(1)$ equations

$$\textstyle\sum_{P \in S_\alpha} e_{(a,b),P} \mathcal{X}_P + \sum_{j=1}^{r_\alpha} \lambda_j(a - b\alpha)\mathcal{X}_j \equiv$$

$$\textstyle\sum_{u \in U} e_{(a,b),u} \log_g \overline{u} + \sum_{Q \in S_\beta} e_{(a,b),Q} \log_g \overline{\gamma_Q} \pmod{q}$$

in unknowns $\mathcal{X}_P$, $\mathcal{X}_j$, $\log_g \overline{\gamma_Q}$ and $\log_g \overline{u}$. This sparse system is solved for its unknowns in time $L_p[s_\alpha, 2c_\alpha] + L_p[s_\beta, 2c_\beta]$, using a sparse matrix technique. In order to get a unique non-zero solution of the system, we set $\log_g \overline{\gamma_Q} = 1$ for a $Q \in S_\beta$ such that $\overline{\gamma_Q}$ is a generator in $\mathbb{F}_p^*$. This ends the pre-computation stage. The running time for this stage is optimal for parameters $s_\alpha = s_\beta = s_d = s_l = \frac{1}{3}$, $c_\alpha = c_\beta = c_l = \left(\frac{8}{9}\right)^{1/3}$, $c_d = \left(\frac{3}{8}\right)^{1/3}$ and then equals $L_p[\frac{1}{3}, \left(\frac{64}{9}\right)^{1/3}]$.

Set $K = L_p[s_k, c_k]$. To find an individual logarithm $\log_a b \pmod{p-1}$ for $a, b \in \mathbb{F}_p^*$ and a a generator of $\mathbb{F}_p^*$, the following procedure for $y = a$ and $y = b$ is executed. Let s be the largest small prime whose logarithm can be computed from the factor bases. Set $z = s^i y \mod p$ for $i = 1$. (Increase i if no good representation can be found.) Use lattice basis reduction to find quotients

$$z \equiv \frac{a_0 + a_1\mu + \cdots + a_d\mu^d}{b_0 + b_1\mu + \cdots + b_d\mu^d} \pmod{p}, \qquad (18)$$

where $a_0, a_1, \ldots, a_d, b_0, b_1, \ldots, b_d$ are integers of size $O(p^{1/(2d+2)})$ such that $\gcd(a_0, a_1, \ldots, a_d) = \gcd(b_0, b_1, \ldots, b_d) = 1$. Check whether both $|\mathrm{Norm}(a_0 + a_1\beta + \cdots + a_d\beta^d)\;|$ and $|\mathrm{Norm}(b_0 + b_1\beta + \cdots + b_d\beta^d)\;|$ are coprime with the index $[\vartheta_\beta, \mathbb{Z}[\beta]]$ and K-smooth, using a $L_p[\frac{s_k}{2}, \sqrt{2s_kc_k}]$-costing ECM-test. From Proposition 2 of [21], applied for $h_1(X) = a_0 + a_1X + \cdots + a_dX^d$ and $h_2(X) = b_0 + b_1X + \cdots + b_dX^d$, it follows that both norms are $\leq L_p[s_d, \frac{3s_dc_d}{2}]L_p[1, \frac{1}{2}] = L_p[1, \frac{1}{2}]$. Using Theorem 1, we see that the probability for these numbers to be simultaneously K-smooth is $L_p[1 - s_k, -\frac{1-s_k}{c_k}]$. Since the lattice-reduction only costs time $L_p[0, 3]$, we conclude that finding a good representation of z takes time $L_p[1-s_k, \frac{1-s_k}{c_k}]L_p[\frac{s_k}{2}, \sqrt{2s_kc_k}]$, which is minimal for $s_k = 2/3$, $c_k = (1/3)^{1/3}$ and then equals $L_p[\frac{1}{3}, 3^{1/3}]$. We show that the time needed to execute the rest of the individual logarithm algorithm is less.

One can easily show that the ideals $(a_0 + a_1\beta + \cdots + a_d\beta^d)\vartheta_\beta$ and $(b_0 + b_1\beta + \cdots + b_d\beta^d)\vartheta_\beta$ split completely into first degree prime ideals. Thus,

$$(a_0 + a_1\beta + \cdots + a_d\beta^d)\vartheta_\beta = \textstyle\prod_{Q \in \tilde{S}_\beta} Q^{v_Q} \;,$$
$$(b_0 + b_1\beta + \cdots + b_d\beta^d)\vartheta_\beta = \textstyle\prod_{Q \in \tilde{S}_\beta} Q^{w_Q} \;,$$

for $\tilde{S}_\beta$ a set of degree one prime ideals in ϑ_β with norm less then K. These equalities imply the equations

$$\begin{aligned}\log_g(a_0 + a_1\mu + \cdots + a_d\mu^d) &\equiv \textstyle\sum_{u\in U} e_{v,u}\log_g \overline{u} + \sum_{Q\in\tilde{S}_\beta} v_Q \log_g \overline{\gamma_Q} (\bmod\ q)\ ,\\ \log_g(b_0 + b_1\mu + \cdots + b_d\mu^d) &\equiv \textstyle\sum_{u\in U} e_{w,u}\log_g \overline{u} + \sum_{Q\in\tilde{S}_\beta} w_Q \log_g \overline{\gamma_Q} (\bmod\ q)\ .\end{aligned}$$

Remark that $\log_g \overline{\gamma_Q}$ is unknown for all $Q \in \tilde{S}_\beta \setminus S_\beta$. To find these unknown logarithms, we reduce the ideal Q in a similar way as described above, searching numbers a, b in an appropriate lattice such that $\mid b^{d+1} f_\beta(a/b) \mid /\mathrm{Norm}(Q)(\in \mathbb{Z})$ and $\mid b^d f_\alpha(a/b) \mid$ are simultaneously $\mathrm{Norm}(Q)^\nu$- smooth for a $\nu < 1$. Medium-sized prime ideals at the f_α-side are reduced similarly. One can check that the asymptotical running time for the reduction of prime ideals Q (at any side) with $L_p[\frac{1}{2}, c_m] < \mathrm{Norm}(Q) \leq L_p[\frac{2}{3}, (\frac{1}{3})^{1/3}]$ is minimal for $\nu = (1/2)^{2/3}$ and then equals $L_p[\frac{1}{3}, (\frac{3}{2})^{1/3}]$. By taking $\nu \geq (4 + 4^{1/3})/256^{1/3}$, time for the reduction of an ideal Q (at any side) with $B_\alpha = B_\beta < \mathrm{Norm}(Q) \leq L_p[\frac{1}{2}, c_m]$ is less then $L_p[\frac{1}{3}, (\frac{3}{2})^{1/3}]$, where c_m is a constant. Following an analogous reasoning as in Section 4.3, one can then see that all unknown $\log_g \overline{\gamma_Q}$ in the above equalities can be determined in time $L_p[\frac{1}{3}, (\frac{3}{2})^{1/3}]$.

Finally, compute $\log_g y$ as

$$\log_g y \equiv -i \log_g s + \log_g(a_0 + a_1\mu + \cdots + a_d\mu^d) - \log_g(b_0 + b_1\mu + \cdots + b_d\mu^d)\ (\bmod\ q)\ ,$$

(see (18)) and then determine the asked for $\log_a b \ (\bmod\ p-1)$ in the same way as in the former individual logarithm algorithm, thus costing time $O(\ln^3 p)$.

We conclude that a seperate individual logarithm stage takes asymptotic time $L_p[\frac{1}{3}, 3^{1/3}]$, after a $L_p[\frac{1}{3}, (\frac{64}{9})^{1/3}]$-costing pre-computation stage .

Acknowledgement

The paper was partially written when Professor I.Semaev was staying at the Department of Mathematics, Section Algebra, Catholic University of Leuven under the project Flanders FWO G.0186.02.
We want to thank the anonymous referees for their very detailed and valuable comments.

References

1. Canfield, E., Erdös, P., Pomerance, C.: On a problem of Oppenheim concerning "factorisatio numerorum". J.Number Theory **17** (1983) 1–28
2. Coppersmith, D.: Fast Evaluation of Logarithms in Fields of Characteristic Two. IEEE Transactions on Information Theory IT-30 (1984) 587–594
3. Coppersmith, D.: Modifications to the Number Field Sieve. J. Cryptology **6** (1993) 169–180
4. Coppersmith, D., Odlyzko, A., Schroeppel, R.: Discrete logarithms in $GF(p)$. Algorithmica **1** (1986) 1–15

5. Gordon, D.: Discrete logarithms in $GF(p)$ using the number field sieve. SIAM Journal of Discrete Mathematics **6** (1993) 124–138
6. Joux, A., Lercier, R.: Improvements to the general Number Field Sieve for discrete logarithms in prime fields. Mathematics of Computation **72** (2003) 953–967
7. Joux, A., Lercier, R.: Calcul de logarithmes discrets dans $GF(p)$ — 130 chiffres. CRYPTO Mailing List (6/2005)
8. Lenstra, A., Lenstra, H. (eds): The Development of the Number Field Sieve. Lecture Notes in Mathematics **1554** , Springer-Verlag, 1993
9. Lenstra, H.: Factoring integers with elliptic curves. Annals of Mathematics **126** (1987) 649–673
10. Matyukhin, D.: On asymptotic complexity of computing discrete logarithms over $GF(p)$. Discrete Mathematics and Applications **13** (2003) 27–50
11. McCurley, K.: The discrete logarithm problem, in: Pomerance,C. (ed): Cryptography and Computational Number Theory. Proc. Symp.Appl.Math. **42**, Amer. Math. Soc.,1990, 49–74
12. Odlyzko, A.: Discrete logarithms: The past and the future. Designs, Codes and Cryptography **19** (2000), 129–145.
13. Odlyzko, A.: Discrete Logarithms and Smooth Polynomials, in: Mullen, G., Shiue, P. (eds): Finite Fields: Theory, Applications and Algorithms. Contemporary Math **168**, Amer. Math. Soc.,1994, 269–278
14. Odlyzko, A.: Discrete logarithms in finite fields and their cryptographic significance, in: Beth, T.,Cot, N., Ingemarsson, I. (eds): Advances in Cryptology: Proceedings of Eurocrypt '84. Lecture Notes in Computer Science **208**, Springer-Verlag,1985,224–314
15. Odlyzko, A.: On the complexity of Computing Discrete Logarithms and Factoring Integers, in: Cover,T. and Gopinath,B. (eds.): Open Problems in Communication and Computation. Springer, 1987, 113-116
16. Pollard, J.: Monte Carlo methods for index computations mod p. Mathematics of Computation **32** (1978) 918–924
17. Pollard, J.: Factoring with cubic integers, in:[8].Springer-Verlag, 1993, 4–10
18. Pomerance, C.: Fast, rigorous factorization and discrete logarithm algorithms, in: Nozaki, N., Johnson, D., Nishizaki, T.,Wilf, H.(eds): Discrete Algorithms and Complexity. Academic Press, 1987, 119–143
19. Schirokauer, O.: Discrete logarithms and local units. Philosophical Transactions of the Royal Society of London (A) **345** (1993) 409–423
20. Schirokauer, O.: Virtual Logarithms. Journal of Algorithms **57** (2005) 140–147
21. Semaev, I.: Special prime numbers and discrete logs in prime finite fields. Mathematics of Computation **71** (2002) 363–377
22. Shoup, V.: Searching for primitive roots in finite fields. Mathematics of Computation **58** (1992) 918–924
23. van Oorschot, P., Wiener, M.: Parallel collision search with cryptanalytic applications. J. Cryptology **12** (1999) 1–28
24. Wiedemann, D.: Solving sparse linear equations over finite fields. IEEE Trans.Inform. Theory **32** (1986) 54–62