Computing Individual Discrete Logarithms Faster in GF(p^n) with the NFS-DL Algorithm

Aurore Guillevic

Computing Individual Discrete Logarithms Faster in GF(p^n) with the NFS-DL Algorithm ⋆ ⋆⋆ ⋆⋆⋆

Aurore Guillevic[1,2]

[1] Institut National de Recherche en Informatique et en Automatique (INRIA)
Grace Team, Inria Saclay, Palaiseau, France
[2] École Polytechnique/LIX, Palaiseau, France
guillevic@lix.polytechnique.fr

Abstract. The Number Field Sieve (NFS) algorithm is the best known method to compute discrete logarithms (DL) in finite fields $\mathbb{F}_{p^n}$, with p medium to large and $n \geq 1$ small. This algorithm comprises four steps: polynomial selection, relation collection, linear algebra and finally, individual logarithm computation. The first step outputs two polynomials defining two number fields, and a map from the polynomial ring over the integers modulo each of these polynomials to $\mathbb{F}_{p^n}$. After the relation collection and linear algebra phases, the (virtual) logarithm of a subset of elements in each number field is known. Given the target element in $\mathbb{F}_{p^n}$, the fourth step computes a preimage in one number field. If one can write the target preimage as a product of elements of known (virtual) logarithm, then one can deduce the discrete logarithm of the target.

As recently shown by the Logjam attack, this final step can be critical when it can be computed very quickly. But we realized that computing an individual DL is much slower in medium- and large-characteristic non-prime fields $\mathbb{F}_{p^n}$ with $n \geq 3$, compared to prime fields and quadratic fields $\mathbb{F}_{p^2}$. We optimize the first part of individual DL: the *booting step*, by reducing dramatically the size of the preimage norm. Its smoothness probability is higher, hence the running-time of the booting step is much improved. Our method is very efficient for small extension fields with $2 \leq n \leq 6$ and applies to any $n > 1$, in medium and large characteristic.

Keywords: Discrete logarithm, finite field, number field sieve, individual logarithm.

⋆⋆ This research was partially funded by Agence Nationale de la Recherche grant ANR-12-BS02-0001.

⋆⋆⋆ Publisher version September, 7th 2015, revised May, 26th 2016.

1 Introduction

1.1 Cryptographic Interest

Given a cyclic group $(G, \cdot)$ and a generator g of G, the discrete logarithm (DL) of $x \in G$ is the element $1 \leq a \leq \#G$ such that $x = g^a$. In well-chosen groups, the exponentiation $(g, a) \mapsto g^a$ is very fast but computing a from (g, x) is conjectured to be very difficult: this is the Discrete Logarithm Problem (DLP), at the heart of many asymmetric cryptosystems. The first group proposed for DLP was the multiplicative group of a prime finite field. Nowadays, the group of points of elliptic curves defined over prime fields are replacing the prime fields for DLP-based cryptosystems. In pairing-based cryptography, the finite fields are still used, because they are a piece in the pairing mechanism. It is important in cryptography to know precisely the difficulty of DL computation in the considered groups, to estimate the security of the cryptosystems using them. Finite fields have a particularity: there exists a subexponential-time algorithm to compute DL in finite fields of medium to large characteristic: the Number Field Sieve (NFS). In small characteristic, this is even better: a quasi-polynomial-time algorithm was proposed very recently [7].

In May 2015, an international team of academic researchers revealed a surprisingly efficient attack against a Diffie-Hellman key exchange in TLS, the *Logjam* attack [2]. After a seven-day-precomputation stage (for relation collection and linear algebra of NFS-DL algorithm), it was made possible to compute any given individual DL in about one minute, for each of the two targeted 512-bit prime finite fields. This was fast enough for a man-in-the-middle attack. This experience shows how critical it can be to be able to compute individual logarithms very fast.

Another interesting application for fast individual DL is *batch-DLP*, and *delayed-target DLP*: in these contexts, an attacker aims to compute several DL in the same finite field. Since the costly phases of relation collection and linear algebra are only done one time for any fixed finite field, only the time for one individual DL is multiplied by the number of targets. This context usually arises in pairing-based cryptography and in particular in broadcast protocols and traitor tracing schemes, where a large number of DLP-based public/private key pairs are generated. The time to compute one individual DL is important in this context, even if parallelization is available.

1.2 The Number Field Sieve Algorithm for DL in Finite Fields

We recall that the NFS algorithm is made of four steps: polynomial selection, relation collection, linear algebra and finally, individual logarithm computation. *This last step is mandatory to break any given instance of a discrete logarithm problem.* The polynomial selection outputs two irreducible polynomials f and g defining two number fields K_f and K_g. One considers the rings $R_f = \mathbb{Z}[x]/(f(x))$ and $R_g = \mathbb{Z}[x]/(g(x))$. There exist two maps ρ_f, ρ_g to $\mathbb{F}_{p^n}$, as shown in the following diagram. Moreover, the monic polynomial defining the finite field is

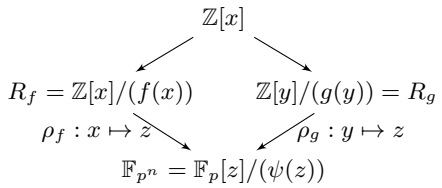

Fig. 1. NFS diagram

$\psi = \gcd(f, g) \bmod p$, of degree n. In the remaining of this paper, we will only use $\rho = \rho_f$, $K = K_f$ and R_f. After the relation collection and linear algebra phases, the (virtual) logarithm of a subset of elements in each ring R_f, R_g is known. The individual DL step computes a preimage in one of the rings R_f, R_g of the target element in $\mathbb{F}_{p^n}$. If one can write the target preimage as a product of elements of known (virtual) logarithm, then one can deduce the individual DL of the target. The key point of individual DL computation is finding a smooth decomposition in small enough factors of the target preimage.

1.3 Previous Work on Individual Discrete Logarithm

The asymptotic running time of NFS algorithm steps are estimated with the L-function:

$$L_Q[\alpha, c] = \exp\Big(\big(c + o(1)\big)(\log Q)^{\alpha}(\log\log Q)^{1-\alpha}\Big) \quad \text{with } \alpha \in [0, 1] \text{ and } c > 0\ .$$

The α parameter measures the gap between polynomial time ($L_Q[\alpha = 0, c] = \log^c Q$) and exponential time ($L_Q[\alpha = 1, c] = Q^c$). When c is implicit, or obvious from the context, we simply write $L_Q[\alpha]$. When the complexity relates to an algorithm for a prime field $\mathbb{F}_p$, we write $L_p[\alpha, c]$.

Large prime fields. Many improvements for computing discrete logarithms first concerned prime fields. The first subexponential DL algorithm in prime fields was due to Adleman [1] and had a complexity of $L_p[1/2, 2]$. In 1986, Coppersmith, Odlyzko and Schroeppel [13] introduced a new algorithm (COS), of complexity $L_p[1/2, 1]$. They computed individual DL [13, §6] in $L_p[1/2, 1/2]$ in two steps (finding a boot of medium-sized primes, then finding relations of logarithms in the database for each medium prime). In these two algorithms, the factor basis was quite large (the smoothness bound was $L_p[1/2, 1/2]$ in both cases), providing a much faster individual DL compared to relation collection and linear algebra. This is where the common belief that individual logarithms are easy to find (and have a negligible cost compared with the prior relation collection and linear algebra phases) comes from.

In 1993, Gordon [15] proposed the first version of NFS–DL algorithm for prime fields $\mathbb{F}_p$ with asymptotic complexity $L_p[1/3, 9^{1/3} \simeq 2.08]$. However, with the $L_p[1/3]$ algorithm there are new difficulties, among them the individual DL

phase. In this $L_p[1/3]$ algorithm, many fewer logarithms of small elements are known, because of a smaller smoothness bound (in $L_p[1/3]$ instead of $L_p[1/2]$). The relation collection is shortened, explaining the $L_p[1/3]$ running time. But in the individual DL phase, since some non-small elements in the decomposition of the target have an unknown logarithm, a dedicated sieving and linear algebra phase is done for each of them. Gordon estimated the running-time of individual DL computation to be $L_p[1/3, 9^{1/3} \simeq 2.08]$, i.e. *the same as the first two phases.* In 1998, Weber [24, §6] compared the NFS–DL algorithm to the COS algorithm for a 85 decimal digit prime and made the same observation about individual DL cost.

In 2003, ten years after Gordon's algorithm, Joux and Lercier [17] were the first to dissociate in NFS relation collection plus linear algebra on one side and individual DL on the other side. They used the *special-q* technique to find the logarithm of medium-sized elements in the target decomposition. In 2006, Commeine and Semaev [11] analyzed the Joux–Lercier method. They obtained an asymptotic complexity of $L_p[1/3, 3^{1/3} \simeq 1.44]$ for computing individual logarithms, independent of the relation collection and linear algebra phases. In 2013, Barbulescu [4, §4, §7.3] gave a tight analysis of the individual DL computation for prime fields, decomposed in three steps: booting (also called smoothing), descent, and final combination of logarithms. The booting step has an asymptotic complexity of $L_p[1/3, 1.23]$ and the descent step of $L_p[1/3, 1.21]$. The final computation has a negligible cost.

Non-prime fields of medium to large characteristic. In 2006, Joux, Lercier, Smart and Vercauteren [19] computed a discrete logarithm in a cubic extension of a prime field. They used the special-q descent technique again. They proposed for large characteristic fields an equivalent of the *rational reconstruction* technique for prime fields and the *Waterloo* algorithm [8] for small characteristic fields, to improve the initializing step preceding the descent. For DLs in prime fields, the target is an integer modulo p. The rational reconstruction method outputs two integers of half size compared to p, such that their quotient is equal to the target element modulo p. Finding a smooth decomposition of the target modulo p becomes equivalent to finding a (simultaneous) smooth decomposition of two elements, each of half the size. We explain their method (that we call the JLSV fraction method in the following) for extension fields in Sec. 2.3.

Link with polynomial selection. The running-time for finding a smooth decomposition depends on the norm of the target preimage. The norm preimage depends on the polynomial defining the number field. In particular, the smaller the coefficients and degree of the polynomial, the smaller the preimage norm. Some polynomial selection methods output polynomials that produce much smaller norm. That may be one of the reasons why the record computation of Joux *et al.* [19] used another polynomial selection method, whose first polynomial has very small coefficients, and the second one has coefficients of size $O(p)$. Thanks to the very small coefficients of the first polynomial, their fraction technique was very useful. Their polynomial selection technique is now superseded by their JLSV_1

method [19, §2.3] for larger values of p. As noted in [19, §3.2], the fraction technique is useful in practice for small n. But for the JLSV$_1$ method and $n \geq 3$, this is already too slow (compared to not using it). In 2008, Zajac [25] implemented the NFS-DL algorithm for computing DLs in $\mathbb{F}_{p^6}$ with p of 40 bits (12 decimal digits (dd), i.e. $\mathbb{F}_{p^6}$ of 240 bits or 74 dd). He used the methods described in [19], with a first polynomial with very small coefficients and a second one with coefficients in $O(p)$. In this case, individual DL computation was possible (see the well-documented [25, §8.4.5]). In 2013, Hayasaka, Aoki, Kobayashi and Takagi [16] computed a DL in $\mathbb{F}_{p^{12}}$ with $p = 122663$ (p^n of 203 bits or 62 dd). We noted that all these records used the same polynomial selection method, so that one of the polynomials has very small coefficients (e.g. $f = x^3 + x^2 - 2x - 1$) whereas the second one has coefficients in $O(p)$.

In 2009, Joux, Lercier, Naccache and Thomé [18] proposed an attack of DLP in a protocol context. The relation collection is sped up with queries to an oracle. They wrote in [18, §B] an extended analysis of individual DL computation. In their case, the individual logarithm phase of the NFS-DL algorithm has a running-time of $L_Q[1/3, c]$ where $c = 1.44$ in the large characteristic case, and $c = 1.62$ in the medium characteristic case. In 2014, Barbulescu and Pierrot [3] presented a multiple number field sieve variant (MNFS) for extension fields, based on Coppersmith's ideas [12]. The individual logarithm is studied in [3, §A]. They also used a *descent* technique, for a global estimated running time in $L_Q[1/3, (9/2)^{1/3}]$, with a constant $c \approx 1.65$. Recently in 2014, Barbulescu, Gaudry, Guillevic and Morain [5,6] announced 160 and 180 decimal digit discrete logarithm records in quadratic fields. They also used a technique derived from the JLSV fraction method and a special-q descent technique, but did not give an asymptotic running-time. It appears that this technique becomes inefficient as soon as $n = 3$ or 4.

Overview of NFS-DL asymptotic complexities. The running-time of the relation collection step and the individual DL step rely on the smoothness probability of integers. An integer is said to be B-smooth if all its prime divisors are less than B. An ideal in a number field is said to be B-smooth if it factors into prime ideals whose norms are bounded by B. Usually, the relation collection and the linear algebra are balanced, so that they have both the same dominating asymptotic complexity. The NFS algorithm for DL in prime and large characteristic fields has a dominating complexity of $L_Q[1/3, (\frac{64}{9})^{1/3} \simeq 1.923]$. For the individual DL in a prime field $\mathbb{F}_p$, the norm of the target preimage in the number field is bounded by p. This bound gives the running time of this fourth step (much smaller than relation collection and linear algebra). Finding a smooth decomposition of the preimage and computing the individual logarithm (see [11]) has complexity $L_p[1/3, c]$ with $c = 1.44$, and $c = 1.23$ with the improvements of [4]. The booting step is dominating. In large characteristic fields, the individual DL has a complexity of $L_Q[1/3, 1.44]$, dominated by the booting step again ([18, §B] for JLSV$_2$, Table 3 for gJL).

In generic medium characteristic fields, the complexity of the NFS algorithm is $L_Q[1/3, (\frac{128}{9})^{1/3} = 2.42]$ with the JLSV$_1$ method proposed in [19, §2.3],

$L_Q[1/3, (\frac{32}{3})^{1/3} = 2.20]$ with the Conjugation method [6], and $L_Q[1/3, 2.156]$ with the MNFS version [23]. We focus on the individual DL step with the JLSV_1 and Conjugation methods. In these cases, the preimage norm bound is in fact much higher than in prime fields. Without any improvements, the dominating booting step has a complexity of $L_Q[1/3, c]$ with $c = 1.62$ [18, §C] or $c = 1.65$ [3, §A]. However, this requires to sieve over ideals of degree $1 < t < n$. For the Conjugation method, this is worse: the booting step has a running-time of $L_Q[1/3, 6^{1/3} \simeq 1.82]$ (see our computations in Table 3). Applying the JLSV fraction method lowers the norm bound to $O(Q)$ for the Conjugation method. The individual logarithm in this case has complexity $L_Q[1/3, 3^{1/3}]$ as for prime fields (without the improvements of [4, §4]). However, this method is not suited for number fields generated with the JLSV_1 method, for $n \geq 3$.

1.4 Our Contributions

In practice, we realized that the JLSV fraction method which seems interesting and sufficient because of the $O(Q)$ bound, is in fact not convenient for the gJL and Conjugation methods for n greater than 3. The preimage norm is much too large, so finding a smooth factorization is too slow by an order of magnitude. We propose a way to lift the target from the finite field to the number field, such that the norm is strictly smaller than $O(Q)$ for the gJL and Conjugation methods:

Theorem 1. *Let $n > 1$ and $s \in \mathbb{F}_{p^n}^*$ a random element (not in a proper subfield of $\mathbb{F}_{p^n}$). We want to compute its discrete logarithm modulo ℓ, where $\ell \mid \Phi_n(p)$, with Φ_n the n-th cyclotomic polynomial. Let K_f be the number field given by a polynomial selection method, whose defining polynomial has the smallest coefficient size, and $R_f = \mathbb{Z}[x]/(f(x))$. Then there exists a preimage $\boldsymbol{r}$ in R_f of some $r \in \mathbb{F}_{p^n}^*$, such that $\log \rho(\boldsymbol{r}) \equiv \log s \pmod{\ell}$ and such that the norm of $\boldsymbol{r}$ in K_f is bounded by $O(Q^e)$, where e is equal to*

1. $1 - \frac{1}{n}$ *for the gJL and Conjugation methods;*
2. $\frac{3}{2} - \frac{3}{2n}$ *for the JLSV_1 method;*
3. $1 - \frac{2}{n}$ *for the Conjugation method, if K_f has a well-chosen quadratic subfield satisfying the conditions of Lemma 3;*
4. $\frac{3}{2} - \frac{5}{2n}$ *for the JLSV_1 method, if K_f has a well-chosen quadratic subfield satisfying the conditions of Lemma 3.*

Our method reaches the optimal bound of $Q^{\varphi(n)/n}$, with $\varphi(n)$ the Euler totient function, for $n = 2, 3, 4, 5$ combined with the gJL or the Conjugation method. We show that our method provides a dramatic improvement for individual logarithm computation for small n: the running-time of the booting step (finding boots) is $L_Q[1/3, c]$ with $c = 1.14$ for $n = 2, 4$, $c = 1.26$ for $n = 3, 6$ and $c = 1.34$ for $n = 5$. It generalizes to any n, so that the norm is always smaller than $O(Q)$ (the prime field case), hence the booting step running-time in $L_Q[1/3, c]$ always satisfies $c < 1.44$ for the two state-of-the-art variants of NFS for extension fields (we have $c = 1.44$ for prime fields). For the JLSV_1 method, this bound is satisfied for $n = 4$, where we have $c = 1.38$ (see Table 3).

1.5 Outline

We select three polynomial selection methods involved for NFS-DL in generic extension fields and recall their properties in Sec. 2.1. We recall a commonly used bound on the norm of an element in a number field (Sec. 2.2). We present in Sec. 2.3 a generalization of the JLSV fraction method of [19]. In Sec. 3.1 we give a proof of the booting step complexity stated in Lemma 1. We sketch in Sec. 3.2 the special-q descent technique and list the asymptotic complexities found in the literature according to the polynomial selection methods. We present in Sec. 4 our main idea to reduce the norm of the preimage in the number field, by reducing the preimage coefficient size with the LLL algorithm. We improve our technique in Sec. 5 by using a quadratic subfield when available, to finally complete the proof of Theorem 1. We provide practical examples in Sec. 6, for 180 dd finite fields in Sec. 6.1 and we give our running-time experiments for a 120 dd finite field $\mathbb{F}_{p^4}$ in Sec. 6.2.

2 Preliminaries

We recall an important property of the LLL algorithm [21] that we will widely use in this paper. Given a lattice $\mathcal{L}$ of $\mathbb{Z}^n$ defined by a basis given in an $n \times n$ matrix L, and parameters $\frac{1}{4} < \delta < 1$, $\frac{1}{2} < \eta < \sqrt{\delta}$, the LLL algorithm outputs a (η, δ)-*reduced basis* of the lattice. the coefficients of the first (shortest) vector are bounded by

$$(\delta - \eta^2)^{\frac{n-1}{4}} \det(L)^{1/n} .$$

With (η, δ) close to $(0.5, 0.999)$ (as in NTL or magma), the approximation factor $C = (\delta - \eta^2)^{\frac{n-1}{4}}$ is bounded by 1.075^{n-1} (see [10, §2.4.2])). Gama and Nguyen experiments [14] on numerous random lattices showed that on average, $C \approx 1.021^n$. In the remaining of this paper, we will simply denote by C this LLL approximation factor.

2.1 Polynomial Selection Methods

We will study the booting step of the NFS algorithm with these three polynomial selection methods:

1. the Joux–Lercier–Smart–Vercauteren (JLSV_1) method [19, §2.3];
2. the generalized Joux–Lercier (gJL) method [22, §2], [6, §3.2];
3. the Conjugation method [6, §3.3].

In a non-multiple NFS version, the JLSV_2 [19, §2.3] and gJL methods have the best asymptotic running-time in the large characteristic case, while the Conjugation method holds the best one in the medium characteristic case. However for a record computation in $\mathbb{F}_{p^2}$, the Conjugation method was used [6]. For medium characteristic fields of record size (between 150 and 200 dd), is seems also that the JLSV_1 method could be chosen ([6, §4.5]). Since the use of each method is not fixed in practice, we study and compare the three above methods for the individual logarithm step of NFS. We recall now the construction and properties of these three methods.

Joux–Lercier–Smart–Vercauteren (JLSV$_1$) Method. This method was introduced in 2006. We describe it in Algorithm 1. The two polynomials f, g have degree n and coefficient size $O(p^{1/2})$. We set $\psi = \gcd(f, g) \bmod p$ monic of degree n. We will use ψ to represent the finite field extension $\mathbb{F}_{p^n} = \mathbb{F}_p[x]/(\psi(x))$.

Algorithm 1: Polynomial selection with the JLSV$_1$ method [19, §2.3]

Input: p prime and n integer
Output: f, g, ψ with $f, g \in \mathbb{Z}[x]$ irreducible and $\psi = \gcd(f \bmod p, g \bmod p)$ in $\mathbb{F}_p[x]$ irreducible of degree n

Select $f_1(x), f_0(x)$, two polynomials with small integer coefficients, $\deg f_1 < \deg f_0 = n$
repeat
  choose $y \approx \lceil \sqrt{p} \rceil$
until $f = f_0 + y f_1$ *is irreducible in* $\mathbb{F}_p[x]$
$(u, v) \leftarrow$ a rational reconstruction of y modulo p
$g \leftarrow v f_0 + u f_1$
return $(f, g, \psi = f \bmod p)$

Generalized Joux–Lercier (gJL) Method. This method was independently proposed in [22, §2] and [4, §8.3] (see also [6, §3.2]). This is a generalization of the Joux–Lercier method [17] for prime fields. We sketch this method in Algorithm 2. The coefficients of g have size $O(Q^{1/(d+1)})$ and those of f have size $O(\log p)$, with $\deg g = d \geq n$ and $\deg f = d + 1$.

Algorithm 2: Polynomial selection with the gJL method

Input: p prime, n integer and $d \geq n$ integer
Output: f, g, ψ with $f, g \in \mathbb{Z}[x]$ irreducible and $\psi = \gcd(f \bmod p, g \bmod p)$ in $\mathbb{F}_p[x]$ irreducible of degree n

Choose a polynomial $f(x)$ of degree $d + 1$ with small integer coefficients which has a monic irreducible factor $\psi(x) = \psi_0 + \psi_1 x + \cdots + x^n$ of degree n modulo p
Reduce the following matrix using LLL

$$M = \begin{bmatrix} p & & & & \\ & \ddots & & & \\ & & p & & \\ \psi_0 & \psi_1 & \cdots & 1 & \\ & \ddots & \ddots & & \ddots \\ & & \psi_0 & \psi_1 & \cdots & 1 \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}a\\a\\a\\a\end{matrix}}\right\} \deg \psi = n \\ \left.\vphantom{\begin{matrix}a\\a\\a\\a\end{matrix}}\right\} d + 1 - n \end{matrix}, \text{ to get } \mathrm{LLL}(M) = \begin{bmatrix} g_0 & g_1 & \cdots & g_d \\ & & & \\ & & * & \\ & & & \end{bmatrix}.$$

return $(f, g = g_0 + g_1 x + \cdots + g_d x^d, \psi)$

Conjugation Method. This method was published in [6] and used for the discrete logarithm record in $\mathbb{F}_{p^2}$, with $f = x^4 + 1$. The coefficient size of f is in $O(\log p)$ and the coefficient size of g is in $O(p^{1/2})$. We describe it in Algorithm 3.

Algorithm 3: Polynomial selection with the Conjugation method [6, §3.3]

Input: p prime and n integer
Output: f, g, ψ with $f, g \in \mathbb{Z}[x]$ irreducible and $\psi = \gcd(f \bmod p, g \bmod p)$ in $\mathbb{F}_p[x]$ irreducible of degree n

repeat
  Select $g_1(x), g_0(x)$, two polynomials with small integer coefficients, $\deg g_1 < \deg g_0 = n$
  Select $P_y(Y)$ a quadratic, monic, irreducible polynomial over $\mathbb{Z}$ with small coefficients
until *$P_y(Y)$ has a root y in $\mathbb{F}_p$ and $\psi(x) = g_0(x) + y g_1(x)$ is irreducible in $\mathbb{F}_p[x]$*
$f \leftarrow \mathrm{Res}_Y(P_y(Y), g_0(x) + Y g_1(x))$
$(u, v) \leftarrow$ a rational reconstruction of y
$g \leftarrow v g_0 + u g_1$
return (f, g, ψ)

Table 1. Properties: degree and coefficient size of the three polynomial selection methods for NFS-DL in $\mathbb{F}_{p^n}$. The coefficient sizes are in $O(X)$. To lighten the notations, we simply write the X term.

method	$\deg f$	$\deg g$	$\|f\|_\infty$	$\|g\|_\infty$
JLSV_1	n	n	$Q^{1/2n}$	$Q^{1/2n}$
gJL	$d+1 > n$	$d \geq n$	$\log p$	$Q^{1/(d+1)}$
Conjugation	$2n$	n	$\log p$	$Q^{1/2n}$

2.2 Norm Upper Bound in a Number Field

In Sec. 4 we will compute the norm of an element s in a number field K_f. We will need an upper bound of this norm. For all the polynomial selection methods chosen, f is monic, whereas g is not. We remove the leading coefficient of f from any formula involved with a monic f. So let f be a monic irreducible polynomial over $\mathbb{Q}$ and let $K_f = \mathbb{Q}[x]/(f(x))$ a number field. Write $s \in K_f$ as a polynomial in x, i.e. $s = \sum_{i=0}^{\deg f - 1} s_i x^i$. The norm is defined by a resultant computation:

$$\mathrm{Norm}_{K_f/\mathbb{Q}}(s) = \mathrm{Res}(f, s) \ .$$

We use Kalkbrener's bound [20, Corollary 2] for an upper bound:

$$|\mathrm{Res}(f, s)| \leq \kappa(\deg f, \deg s) \cdot |\|f\|_\infty^{\deg s} |\|s\|_\infty^{\deg f},$$

where $\kappa(n,m) = \binom{n+m}{n}\binom{n+m-1}{n}$, and $|||f\|_\infty = \max_{0\le i\le \deg f} |f_i|$ the absolute value of the greatest coefficient. An upper bound for $\kappa(n,m)$ is $(n+m)!$. We will use the following bound in Sec. 4:

$$|\operatorname{Norm}_{K_f/\mathbb{Q}}(s)| \le (\deg f + \deg s)!|||f\|_\infty^{\deg s}|||s\|_\infty^{\deg f} \ . \tag{1}$$

2.3 Joux–Lercier–Smart–Vercauteren Fraction Method

Notation 1 *Row and column indices. In the following, we will define matrices of size $d \times d$, with $d \ge n$. For ease of notation, we will index the rows and columns from 0 to $d-1$ instead of 1 to d, so that the $(i+1)$-th row at index i, $L_i = [L_{ij}]_{0\le j\le d-1}$, can be written in polynomial form $\sum_{j=0}^{d-1} L_{ij}x^j$, and the column index j coincides with the degree j of x^j.*

In 2006 was proposed in [19] a method to generalize to non-prime fields the rational reconstruction method used for prime fields. In the prime field setting, the target is an integer modulo p. The rational reconstruction method outputs two integers of half size compared to p and such that their quotient is equal to the target element modulo p. Finding a smooth decomposition of the target modulo p becomes equivalent to finding at the same time a smooth decomposition of two integers of half size each.

To generalize to extension fields, one writes the target preimage as a quotient of two number field elements, each with a smaller norm compared to the original preimage. We denote by s the target in the finite field $\mathbb{F}_{p^n}$ and by $\boldsymbol{s}$ a preimage (or lift) in K. Here is a first very simple preimage choice. Let $\mathbb{F}_{p^n} = \mathbb{F}_p[x]/(\psi(x))$ and $s = \sum_{i=0}^{\deg s} s_i x^i \in \mathbb{F}_{p^n}$, with $\deg s < n$. We lift the coefficients $s_i \in \mathbb{F}_p$ to $\boldsymbol{s}_i \in \mathbb{Z}$ then we set a preimage of s in the number field K to be

$$\boldsymbol{s} = \sum_{i=0}^{\deg s} \boldsymbol{s}_i X^i \ ,$$

with X such that $K = \mathbb{Q}[X]/(f(X))$. (We can also write $\boldsymbol{s} = \sum_{i=0}^{\deg s} \boldsymbol{s}_i \alpha^i$, with α a root of f in the number field: $K = \mathbb{Q}[\alpha]$). We have $\rho(\boldsymbol{s}) = s$.

Now LLL is used to obtain a quotient whose numerator and denominator have smaller coefficients. We present here the lattice used with the JLSV_1 polynomial selection method. The number field K is of degree n. We define a lattice of dimension $2n$. For the corresponding matrix, each column of the left half corresponds to a power of X in the numerator; each column of the right half corresponds to a power of X in the denominator. The matrix is

$$L = \begin{bmatrix} p & & & & & \\ & \ddots & & & & \\ & & p & & & \\ \boldsymbol{s}_0 & \dots & \boldsymbol{s}_{n-1} & 1 & & \\ \vdots & & \vdots & & \ddots & \\ \boldsymbol{s}x^{n-1} & & (\bmod\ \psi) & & & 1 \end{bmatrix} \begin{matrix} 0 \\ \vdots \\ n-1 \\ n \\ \vdots \\ 2n-1 \end{matrix} \quad {\scriptstyle 2n\times 2n}$$

The first n coefficients of the output vector, $u_0, u_1, \ldots, u_{n-1}$ give a numerator u and the last n coefficients give a denominator v, so that $\boldsymbol{s} = a\frac{u(X)}{v(X)}$ with a a scalar in $\mathbb{Q}$. The coefficients u_i, v_i are bounded by $\|u\|_\infty,\ \|v\|_\infty \leq Cp^{1/2}$ since the matrix determinant is $\det L = p^n$ and the matrix is of size $2n \times 2n$. However the product of the norms of each u, v in the number field K will be much larger than the norm of the single element $\boldsymbol{s}$ because of the large coefficients of f in the norm formula. We use formula (1) to estimate this bound:

$$\mathrm{Norm}_{K/\mathbb{Q}}(u) \leq \|u\|_\infty^{\deg f} \|f\|_\infty^{\deg u} = O(p^{\frac{n}{2}} p^{\frac{n-1}{2}}) = O(p^{n-\frac{1}{2}}) = O(Q^{1-\frac{1}{2n}})$$

and the same for $\mathrm{Norm}_{K/\mathbb{Q}}(v)$, hence the product of the two norms is bounded by $O(Q^{2-\frac{1}{n}})$. The norm of $\boldsymbol{s}$ is bounded by $\mathrm{Norm}_{K/\mathbb{Q}}(\boldsymbol{s}) \leq p^n p^{\frac{n-1}{2}} = Q^{\frac{3}{2}-\frac{1}{2n}}$ which is much smaller whenever $n \geq 3$. Finding a smooth decomposition of u and v at the same time will be much slower than finding one for $\boldsymbol{s}$ directly, for large p and $n \geq 3$. This is mainly because of the large coefficients of f (in $O(p^{1/2})$).

Application to gJL and Conjugation Method. The method of [19] to improve the smoothness of the target norm in the number field K_f has an advantage for the gJL and Conjugation methods. First we note that the number field degree is larger than n: this is $d + 1 \geq n + 1$ for the gJL method and $2n$ for the Conjugation method. For ease of notation, we denote by d_f the degree of f. We define a lattice of dimension $2d_f$. Hence there is more place to reduce the coefficient size of the target $\boldsymbol{s}$.

We put p on the diagonal of the first $n-1$ rows, then $x^i\psi(x)$ coefficients from row n to $d_f - 1$, where $0 \leqslant i < d_f - 1$ (ψ is of degree n and has $n+1$ coefficients). The rows from index d_f to $2d_f$ are filled with $X^i\boldsymbol{s} \bmod f$ (these elements have d_f coefficients). We obtain a triangular matrix L.

$$L = \begin{bmatrix} p & & & & & & & \\ & \ddots & & & & & & \\ & & p & & & & & \\ \psi_0 & \cdots & \psi_{n-1} & 1 & & & & \\ & \ddots & & \ddots & \ddots & & & \\ & & \psi_0 & \cdots & \psi_{n-1} & 1 & & \\ \boldsymbol{s}_0 & \ldots & \boldsymbol{s}_{n-1} & & & & 1 & \\ \vdots & & & & & & & \ddots \\ X^{d_f-1}\boldsymbol{s} \bmod f & & & & & & & 1 \end{bmatrix} \begin{matrix} 0 \\ \vdots \\ n-1 \\ n \\ \vdots \\ d_f-1 \\ d_f \\ \vdots \\ 2d_f \end{matrix}$$
$$2d_f \times 2d_f$$

Since the determinant is $\det L = p^n$ and the matrix of dimension $2d_f \times 2d_f$, the coefficients obtained with LLL will be bounded by $Cp^{\frac{n}{2d_f}}$. The norm of the numerator or the denominator (with $\boldsymbol{s} = u(X)/v(X) \in K_f$) is bounded by

$$\mathrm{Norm}_{K_f/\mathbb{Q}}(u) \leq \|u\|_\infty^{\deg f} \|f\|_\infty^{\deg u} = O(p^{n/2}) = O(Q^{1/2})\ .$$

The product of the two norms will be bounded by $O(Q)$ hence we will have the same asymptotic running time as for prime fields, for finding a smooth decomposition of the target in a number field obtained with the gJL or Conjugation method. We will show in Sec. 4 that we can do even better.

3 Asymptotic Complexity of Individual DL Computation

3.1 Asymptotic Complexity of Initialization or Booting Step

In this section, we prove the following lemma on the booting step running-time to find a smooth decomposition of the norm preimage. This was already proven especially for an initial norm bound of $O(Q)$. We state it in the general case of a norm bound of Q^e. The smoothness bound $B = L_Q[2/3, \gamma]$ used here is not the same as for the relation collection step, where the smoothness bound was $B_0 = L_Q[1/3, \beta_0]$. Consequently, the special-q output in the booting step will be bounded by B.

Lemma 1 (Running-time of B-smooth decomposition). *Let $s \in \mathbb{F}_Q$ of order ℓ. Take at random $t \in [1, \ell - 1]$ and assume that the norm S_t of a preimage of $s^t \in \mathbb{F}_Q$, in the number field K_f, is bounded by $Q^e = L_Q[1, e]$. Write $B = L_Q[\alpha_B, \gamma]$ the smoothness bound for S_t. Then the lower bound of the expected running time for finding t s.t. the norm S_t of s^t is B-smooth is $L_Q[1/3, (3e)^{1/3}]$, obtained with $\alpha_B = 2/3$ and $\gamma = (e^2/3)^{1/3}$.*

First, we need a result on smoothness probability. We recall the definition of B-smoothness already stated in Sec. 1.4: an integer S is B-smooth if and only if all its prime divisors are less than or equal to B. We also recall the L-notation widely used for sub-exponential asymptotic complexities:

$$L_Q[\alpha, c] = \exp\Big(\big(c + o(1)\big)(\log Q)^{\alpha} (\log\log Q)^{1-\alpha} \Big) \quad \text{with } \alpha \in [0, 1] \text{ and } c > 0 \ .$$

The Canfield–Erdős–Pomerance [9] theorem provides a useful result to measure smoothness probability:

Theorem 2 (B-smoothness probability). *Suppose $0 < \alpha_B < \alpha_S \leq 1$, $\sigma > 0$, and $\beta > 0$ are fixed. For a random integer S bounded by $L_Q[\alpha_S, \sigma]$ and a smoothness bound $B = L_Q[\alpha_B, \beta]$, the probability that S is B-smooth is*

$$\Pr(S \textit{ is } B\textit{-smooth}) = L_Q\Big[\alpha_S - \alpha_B, -(\alpha_S - \alpha_B)\frac{\sigma}{\beta}\Big] \tag{2}$$

for $Q \to \infty$.

We prove now the Lemma 1 that states the running-time of individual logarithm when the norm of the target in a number field is bounded by $O(Q^e)$.

Proof (of Lemma 1). From Theorem 2, the probability that S bounded by $Q^e = L_Q[1, e]$ is B-smooth with $B = L_Q[\alpha_B, \gamma]$ is $\Pr(S \text{ is } B\text{-smooth}) = L_Q\big[1 -$

$\alpha_B, -(1-\alpha_B)\frac{e}{\gamma}\Big]$. We assume that a B-smoothness test with ECM takes time $L_B[1/2, 2^{1/2}] = L_Q[\frac{\alpha_B}{2}, (2\gamma\alpha_B)^{1/2}]$. The running-time for finding a B-smooth decomposition of S is the ratio of the time per test (ECM cost) to the B-smoothness probability of S:

$$L_Q\Big[\frac{\alpha_B}{2}, (2\gamma\alpha_B)^{1/2}\Big] L_Q\Big[1-\alpha_B, (1-\alpha_B)\frac{e}{\gamma}\Big] \ .$$

We optimize first the α value, so that $\alpha \leq 1/3$ (that is, not exceeding the α of the two previous steps of the NFS algorithm): $\max(\alpha_B/2, 1-\alpha_B) \leq \frac{1}{3}$. This gives the system $\begin{cases} \alpha_B \leq 2/3 \\ \alpha_B \geq 2/3 \end{cases}$ So we conclude that $\alpha_B = \frac{2}{3}$. The running-time for finding a B-smooth decomposition of S is therefore

$$L_Q\Big[1/3, \Big(\frac{4}{3}\gamma\Big)^{1/2} + \frac{e}{3\gamma}\Big] \ .$$

The minimum[3] of the function $\gamma \mapsto (\frac{4}{3}\gamma)^{1/2} + \frac{e}{3\gamma}$ is $(3e)^{1/3}$, corresponding to $\gamma = (e^2/3)^{1/3}$, which yields our optimal running time, together with the special-q bound B:

$$L_Q\Big[1/3, (3e)^{1/3}\Big] \quad \text{with } q \leq B = L_Q\Big[2/3, (e^2/3)^{1/3}\Big] \ .$$

□

3.2 Running-Time of Special-q Descent

The second step of the individual logarithm computation is the *special-q descent*. This consists in computing the logarithms of the medium-sized elements in the factorization of the target in the number field. The first special-q is of order $L_Q[2/3, \gamma]$ (this is the boot obtained in the initialization step) and is the norm of a degree one prime ideal in the number field where the booting step was done (usually K_f). The idea is to *sieve* over linear combinations of degree one ideals, in K_f and K_g at the same time, whose norms for one side will be multiples of q by construction, in order to obtain a relation involving a degree one prime ideal of norm q and other degree one prime ideals of norm strictly smaller than q.

Here is the common way to obtain such a relation. Let $\mathfrak{q}$ be a degree one prime ideal of K_f, whose norm is q. We can write $\mathfrak{q} = \langle q, r_q \rangle$, with r_q a root of f modulo q (hence $|r_q| < q$). We need to compute two ideals $\mathfrak{q}_1, \mathfrak{q}_2 \in K_f$ whose respective norm is a multiple of q, and sieve over $a\mathfrak{q}_1 + b\mathfrak{q}_2$. The classical way to construct these two ideals is to reduce the two-dimensional lattice generated by q and $r_q - \alpha_f$, i.e. to compute $\mathrm{LLL}\left(\begin{bmatrix} q & 0 \\ -r & 1 \end{bmatrix}\right) = \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \end{bmatrix}$ to obtain two

[3] One computes the derivative of the function $h_{a,b}(x) = a\sqrt{x} + \frac{b}{x}$: this is $h'_{a,b}(x) = \frac{a}{2\sqrt{x}} - \frac{b}{x^2}$ and find that the minimum of h for $x > 0$ is $h_{a,b}((\frac{2b}{a})^{2/3}) = 3(\frac{a^2 b}{4})^{1/3}$. With $a = 2/3^{1/2}$ and $b = e/3$, we obtain the minimum: $h((\frac{e^2}{3})^{1/3}) = (3e)^{1/3}$.

degree-one ideals $u_1 + v_1\alpha_f, u_2 + v_2\alpha_f$ with shorter coefficients. One sieves over $\mathfrak{r}_f = (au_1 + bu_2) + (av_1 + bv_2)\alpha_f$ and $\mathfrak{r}_g = (au_1 + bu_2) + (av_1 + bv_2)\alpha_g$. The new ideals obtained in the relations will be treated as new special-qs until a relation of ideals of norm bounded by B_0 is found, where B_0 is the bound on the factor basis, so that the individual logarithms are finally known. The sieving is done in three stages, for the three ranges of parameters.

1. For $q = L_Q[2/3, \beta_1]$: large special-q;
2. For $q = L_Q[\lambda, \beta_2]$ with $1/3 < \lambda < 2/3$: medium special-q;
3. For $q = L_Q[1/3, \beta_3]$: small special-q.

The proof of the complexity is not trivial at all, and since this step is allegedly cheaper than the two main phases of sieving and linear algebra, whose complexity is $L_Q[1/3, (\frac{64}{9})^{1/3}]$, the proofs are not always expanded.

There is a detailed proof in [11, §4.3] and [4, §7.3] for prime fields $\mathbb{F}_p$. We found another detailed proof in [18, §B] for large characteristic fields $\mathbb{F}_{p^n}$, however this was done for the polynomial selection of [19, §3.2] (which has the same main asymptotic complexity $L_Q[1/3, (\frac{64}{9})^{1/3}]$). In [22, §4, pp. 144–150] the NFS-DL algorithm is not proposed in the same order: the booting and descent steps (step (5) of the algorithm in [22, §2]) are done as a first sieving, then the relations are added to the matrix that is solved in the linear algebra phase. What corresponds to a booting step is proved to have a complexity bounded by $L_Q[1/3, 3^{1/3}]$ and there is a proof that the descent phase has a smaller complexity than the booting step. There is a proof for the JLSV_1 polynomial selection in [18, §C] and [3, §A] for a MNFS variant. We summarize in Tab. 2 the asymptotic complexity formulas for the booting step and the descent step that we found in the available papers.

Table 2. Complexity of the booting step and the descent step for computing one individual DL, in $\mathbb{F}_p$ and $\mathbb{F}_{p^n}$, in medium and large characteristic. The complexity is given by the formula $L_Q[1/3, c]$, only the constant c is given in the table for ease of notation. The descent of a medium special-q, bounded by $L_Q[\lambda, c]$ with $1/3 < \lambda < 2/3$, is proven to be negligible compared to the large and small special-q descents. In [18, §B,C], the authors used a sieving technique over ideals of degree $t > 1$ for large and medium special-q descent.

reference	finite field	polynomial selection	target norm bound	booting step	descent step		
					large	med.	small
[11, §4.3]	$\mathbb{F}_p$	JL03 [17]	p	1.44	<1.44		
[4, Tab. 7.1]	$\mathbb{F}_p$	JL03 [17]	p	1.23	1.21	neg.	0.97
[22, §4]	$\mathbb{F}_{p^n}$, large p	gJL	Q	1.44	< 1.44		
[18, §B]	$\mathbb{F}_{p^n}$, large p	JLSV_2	Q	1.44	–	neg.	1.27
[18, §C]	$\mathbb{F}_{p^n}$, med. p	JLSV_1 variant	$Q^{1+\alpha}$, $\alpha \simeq 0.4$	1.62	–	neg.	0.93
[3, §A]	$\mathbb{F}_{p^n}$, med. p	JLSV_1	$Q^{3/2}$	1.65	≤ 1.03		

Usually, the norm of the target is assumed to be bounded by Q (this is clearly the case for prime fields $\mathbb{F}_p$). The resulting initialization step (finding a boot

for the descent) has complexity $L_Q[1/3, 3^{1/3} \approx 1.44]$. Since the large special-q descent complexity depends on the size of the largest special-q of the boot, lowering the norm, hence the booting step complexity *and* the largest special-q of the boot also decrease the large special-q descent step complexity. It would be a considerable project to rewrite new proofs for each polynomial selection method, according to the new booting step complexities. However, its seems to us that by construction, the large special-q descent step in these cases has a (from much to slightly) smaller complexity than the booting step. The medium special-q descent step has a negligible cost in the cases considered above. Finally, the small special-q descent step does not depend on the size of the boot but on the polynomial properties (degree, and coefficient size). We note that for the JLSV_2 polynomial selection, the constant of the complexity is 1.27. It would be interesting to know the constant for the gJL and Conjugation methods.

The third and final step of individual logarithm computation is very fast. It combines all of the logarithms computed before, to get the final discrete logarithm of the target.

4 Computing a Preimage in the Number Field

Our main idea is to compute a preimage in the number field with smaller degree (less than $\deg s$) and/or of coefficients of reduced size, by using the subfield structure of $\mathbb{F}_{p^n}$. We at least have one non-trivial subfield: $\mathbb{F}_p$. In this section, we reduce the size of the coefficients of the preimage. This reduces its norm and give the first part of the proof of Theorem 1. In the following section, we will reduce the degree of the preimage when n is even, completing the proof.

Lemma 2. *Let $s \in \mathbb{F}_{p^n}^* = \sum_{i=0}^{\deg s} s_i x^i$, with $\deg s < n$. Let ℓ be a non-trivial divisor of $\Phi_n(p)$. Let $s' = u \cdot s$ with u in a proper subfield of $\mathbb{F}_{p^n}$. Then*

$$\log s' \equiv \log s \bmod \ell \ . \tag{3}$$

Proof. We start with $\log s' = \log s + \log u$ and since u is in a proper subfield, we have $u^{(p^n-1)/\Phi_n(p)} = 1$, then $u^{(p^n-1)/\ell} = 1$. Hence the logarithm of u modulo ℓ is zero, and $\log s' \equiv \log s \bmod \ell$. □

Example 1 (Monic preimage). Let s' be equal to s divided by its leading term, $s' = \frac{1}{s_{\deg s}} s \in \mathbb{F}_{p^n}$. We have $\log s' \equiv \log s \bmod \ell$.

We assume in the following that the target s is monic since dividing by its leading term does not change its logarithm modulo ℓ.

4.1 Preimage Computation in the JLSV$_1$ Case

Let $s = \sum_{i=0}^{n-1} s_i x^i \in \mathbb{F}_{p^n}$ with $s_{n-1} = 1$. We define a lattice of dimension n by the $n \times n$ matrix

$$L = \begin{bmatrix} p & & & \\ & \ddots & & \\ & & p & \\ \boldsymbol{s}_0 & \dots & \boldsymbol{s}_{n-2} & 1 \end{bmatrix}_{n\times n} \begin{matrix} 0 \\ \vdots \\ n-2 \\ n-1 \end{matrix} \begin{matrix} \Big\} n-1 \text{ rows} \\ \\ \Big\} \text{row } n-1 \text{ with } \boldsymbol{s} \text{ coeffs} \end{matrix}$$

with p on the diagonal for the first $n-1$ rows (from 0 to $n-2$), and the coefficients of the monic element $\boldsymbol{s}$ on row $n-1$. Applying the LLL algorithm to M, we obtain a reduced element $\boldsymbol{r} = \sum_{i=0}^{n-1} \boldsymbol{r}_i X^i \in K_f$ such that

$$\boldsymbol{r} = \sum_{i=0}^{n-1} a_i L_i$$

with L_i the vector defined by the i-th row of the matrix and a_i a scalar in $\mathbb{Z}$. We map this equality in $\mathbb{F}_{p^n}$ with ρ. All the terms cancel out modulo p except the line with $\boldsymbol{s}$:

$$\rho(\boldsymbol{r}) \equiv \rho(a_{n-1}) \cdot \rho(\boldsymbol{s}) = u \cdot s \mod (p, \psi)$$

with $u = \rho(a_{n-1}) \in \mathbb{F}_p$. Hence, by Lemma 2,

$$\log \rho(\boldsymbol{r}) \equiv \log s \bmod \ell \ . \tag{4}$$

Moreover,

$$\|\boldsymbol{r}\|_\infty \leq C p^{(n-1)/n} \ .$$

It is straightforward, using Inequality (1), to deduce that

$$\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = O\big(p^{\frac{3}{2}(n-1)}\big) = O\big(Q^{\frac{3}{2}-\frac{3}{2n}}\big) \ .$$

We note that this first simple improvement applied to the JLSV$_1$ construction is already better than doing nothing: in that case, $\mathrm{Norm}_{K_f/\mathbb{Q}}(s) = O(Q^{\frac{3}{2}-\frac{1}{2n}})$. The norm of $\boldsymbol{r}$ is smaller by a factor of size $Q^{\frac{1}{n}}$. For $n = 2$ we have $\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = O(Q^{\frac{3}{4}})$ but for $n = 3$, the bound is $\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = O(Q)$, and for $n = 4$, $O(Q^{11/8})$. This is already too large. We would like to obtain such a bound, strictly smaller than $O(Q)$, for any n.

4.2 Preimage Computation in the gJL and Conjugation Cases

Let $s = \sum_{i=0}^{n-1} s_i x^i \in \mathbb{F}_{p^n}$ with $s_{n-1} = 1$. In order to present a generic method for both the gJL and the Conjugation methods, we denote by d_f the degree of f. In the gJL case we have $d_f = d + 1 \geq n + 1$, while in the Conjugation case, $d_f = 2n$. We define the $d_f \times d_f$ matrix with p on the diagonal for the first $n - 1$

rows, and the coefficients of the monic element $\boldsymbol{s}$ on row $n-1$. The rows n to d_f are filled with the coefficients of the monic polynomial $x^j\psi$, with $0 \leq j \leq d_f - n$.

$$L = \begin{bmatrix} p & & & & & & \\ & \ddots & & & & & \\ & & p & & & & \\ \boldsymbol{s}_0 & \dots & \boldsymbol{s}_{n-2} & 1 & & & \\ \psi_0 & \psi_1 & \cdots & \psi_{n-1} & 1 & & \\ & \ddots & \ddots & & \ddots & \ddots & \\ & & \psi_0 & \psi_1 & \cdots & \psi_{n-1} & 1 \end{bmatrix} \begin{matrix} 0 \\ \vdots \\ n-2 \\ n-1 \\ n \\ \vdots \\ d_f-1 \end{matrix} \begin{matrix} \Big\} n-1 \text{ rows} \\ \\ \Big\} \text{row } n-1 \text{ with } \boldsymbol{s} \text{ coeffs} \\ \Big\} d_f - n \text{ rows with } \psi \text{ coeffs} \\ \\ \\ \end{matrix}$$
$$d_f \times d_f$$

Applying the LLL algorithm to L, we obtain a reduced element $\boldsymbol{r} = \sum_{i=0}^{d_f-1} \boldsymbol{r}_i X^i \in K_f$ such that $\boldsymbol{r} = \sum_{i=0}^{d_f-1} a_i L_i$ where L_i is the i-th row vector of L and a_i is a scalar in $\mathbb{Z}$. We map this equality into $\mathbb{F}_{p^n}$ with ρ. All the terms cancel out modulo (p, ψ) except the one with $\boldsymbol{s}$ coefficients:

$$\rho(\boldsymbol{r}) \equiv \rho(a_{n-1}) \cdot \rho(\boldsymbol{s}) = u \cdot s \mod (p, \psi)$$

with $u = \rho(a_{n-1}) \in \mathbb{F}_p$. Hence, by Lemma 2,

$$\log \rho(\boldsymbol{r}) \equiv \log s \bmod \ell \ . \tag{5}$$

Moreover,

$$\|\boldsymbol{r}\|_\infty \leq C p^{(n-1)/d_f} \ .$$

It is straightforward, using Inequality (1), to deduce that

$$\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = O\big(p^{n-1}\big) = O\big(Q^{1-1/n}\big) \ .$$

Here we obtain a bound that is *always strictly smaller* than Q for any n. In the next section we show how to improve this bound to $O\big(Q^{1-2/n}\big)$ when n is even and the number field defined by ψ has a well-suited quadratic subfield.

5 Preimages of Smaller Norm with Quadratic Subfields

Reducing the degree of s can reduce the norm size in the number field for the JLSV$_1$ polynomial construction. We present a way to compute $r \in \mathbb{F}_{p^n}$ of degree $n-2$ from $s \in \mathbb{F}_{p^n}$ of degree n in the given representation of $\mathbb{F}_{p^n}$, and r, s satisfying Lemma 2. We need n to be even and the finite field $\mathbb{F}_{p^n}$ to be expressed as a degree-$n/2$ extension of a quadratic extension defined by a polynomial of a certain form. We can define another lattice with r and get a preimage of degree $n-2$ instead of $n-1$ in the number field. This can be interesting with the JLSV$_1$ method. Combining this method with the previous one of Sec. 4 leads to our proof of Theorem 1.

5.1 Smaller Preimage Degree

In this section, we prove that when n is even and $\mathbb{F}_{p^n} = \mathbb{F}_p[X]/(\psi(X))$ has a quadratic base field $\mathbb{F}_{p^2}$ of a certain form, from a random element $s \in \mathbb{F}_{p^n}$ with $s_{n-1} \neq 0$, we can compute an element $r \in \mathbb{F}_{p^n}$ with $r_{n-1} = 0$, and $s = u \cdot r$ with $u \in \mathbb{F}_{p^2}$. Then, using Lemma 2, we will conclude that $\log r \equiv \log s \bmod \ell$.

Lemma 3. *Let $\psi(X)$ be a monic irreducible polynomial of $\mathbb{F}_p[X]$ of even degree n with a quadratic subfield defined by the polynomial $P_y = Y^2 + y_1 Y + y_0$. Moreover, assume that ψ splits over $\mathbb{F}_{p^2} = \mathbb{F}_p[Y]/(P_y(Y))$ as*

$$\begin{aligned} \psi(X) &= (P_z(X) - Y)(P_z(X) - Y^p) \\ \textit{or}\ \ \psi(X) &= (P_z(X) - YX)(P_z(X) - Y^pX) \end{aligned}$$

with P_z monic, of degree $n/2$ and coefficients in $\mathbb{F}_p$. Let $s \in \mathbb{F}_p[X]/(\psi(X))$ a random element, $s = \sum_{i=0}^{n-1} s_i X^i$.

Then there exists $r \in \mathbb{F}_{p^n}$ monic and of degree $n-2$ in X, and $u \in \mathbb{F}_{p^2}$, such that $s = u \cdot r$ in $\mathbb{F}_{p^n}$.

We first give an example for $s \in \mathbb{F}_{p^4}$ then present a constructive proof.

Example 2. Let $P_y = Y^2 + y_1 Y + y_0$ be a monic irreducible polynomial over $\mathbb{F}_p$ and set $\mathbb{F}_{p^2} = \mathbb{F}_p[Y]/(P_y(Y))$. Assume that $Z^2 - YZ + 1$ is irreducible over $\mathbb{F}_{p^2}$ and set $\mathbb{F}_{p^4} = \mathbb{F}_{p^2}[Z]/(Z^2 - YZ + 1)$. Let $\psi = X^4 + y_1X^3 + (y_0 + 2)X^2 + y_1X + 1$ be a monic reciprocal polynomial. By construction, ψ factors over $\mathbb{F}_{p^2}$ into $(X^2 - YX + 1)(X^2 - Y^pX + 1)$ and $\mathbb{F}_p[X]/(\psi(X))$ defines a quartic extension $\mathbb{F}_{p^4}$ of $\mathbb{F}_p$. We have these two representations for $\mathbb{F}_{p^4}$:

$$\begin{array}{l} \mathbb{F}_{p^4} = \mathbb{F}_{p^2}[Z]/(Z^2 - YZ + 1) \\ \quad | \\ \mathbb{F}_{p^2} = \mathbb{F}_p[Y]/(Y^2 + y_1Y + y_0) \\ \quad | \\ \quad \mathbb{F}_p \end{array} \quad \text{and} \quad \begin{array}{l} \mathbb{F}_{p^4} = \mathbb{F}_p[X]/(X^4 + y_1X^3 + (y_0 + 2)X^2 + y_1X + 1) \\ \quad | \\ \quad | \\ \quad | \\ \quad \mathbb{F}_p \end{array}$$

Proof (of Lemma 3). Two possible extension field towers are:

$$\begin{array}{l} \mathbb{F}_{p^n} = \mathbb{F}_{p^2}[Z]/(P_z(Z) - Y) \\ | \\ \mathbb{F}_{p^2} = \mathbb{F}_p[Y]/(P_y(Y)) \\ | \\ \mathbb{F}_p \end{array} \quad \text{and} \quad \begin{array}{l} \mathbb{F}_{p^n} = \mathbb{F}_{p^2}[Z]/(P_z(Z) - YZ) \\ | \\ \mathbb{F}_{p^2} = \mathbb{F}_p[Y]/(P_y(Y)) \\ | \\ \mathbb{F}_p \end{array}$$

We write s in the following representation to emphasize the subfield structure:

$$s = \sum_{i=0}^{n/2-1} (a_{i0} + a_{i1}Y)Z^i \text{ with } a_{ij} \in \mathbb{F}_p \ .$$

1. If $\psi = P_z(Z) - Y$ then we can divide s by $u_{LT} = a_{n/2,0} + a_{n/2,1}Y \in \mathbb{F}_{p^2}$ (the leading term in Z, i.e. the coefficient of $Z^{n/2}$) to make s monic in Z up to a subfield cofactor u_{LT}:

$$\frac{s}{u_{LT}} = \sum_{i=0}^{n/2-2} (b_{i0} + b_{i1}Y)Z^i \quad + Z^{n/2-1} \ ,$$

with the coefficients b_{ij} in the base field $\mathbb{F}_p$, and $b_{i0}+b_{i1}Y = (a_{i0}+a_{i1}Y)/u_{LT}$. Since $P_z(Z) = Y$ and $Z = X$ in $\mathbb{F}_{p^n}$ by construction, we replace Y by $P_z(Z)$ and Z by X to get an expression for s in X:

$$\frac{s}{u_{LT}} = \sum_{i=0}^{n/2-2} (b_{i0} + b_{i1}P_z(X))X^i + X^{n/2-1} = r(X) \ .$$

The degree in X of r is $\deg r = \deg P_z(X)X^{n/2-2} = n-2$ instead of $\deg s = n-1$. We set $u = 1/u_{LT}$. By construction, $u \in \mathbb{F}_{p^2}$. We conclude that $s = ur \in \mathbb{F}_{p^n}$, with $\deg r = n-2$ and $u \in \mathbb{F}_{p^2}$.
2. If $\psi = P_z(Z) - YZ$ then we can divide s by $u_{CT} = a_{00} + a_{01}Y \in \mathbb{F}_{p^2}$ (the constant term in Z) to make the constant coefficient of s to be 1:

$$\frac{s}{u_{CT}} = 1 + \sum_{i=1}^{n/2-1} (b_{i0} + b_{i1}Y)Z^i$$

with $b_{ij} \in \mathbb{F}_p$. Since $P_z(Z) = YZ$ and $Z = X$ in $\mathbb{F}_{p^n}$ by construction, we replace YZ by $P_z(Z)$ and Z by X to get

$$\frac{s}{u_{CT}} = 1 + \sum_{i=1}^{n/2-1} (b_{i0}X^i + b_{i1}P_z(X)X^{i-1}) = r(X) \ .$$

The degree in X of r is $\deg r = \deg P_z(X)X^{n/2-1-1} = n-2$ instead of $\deg s = n-1$. We set $u = 1/u_{CT}$. By construction, $u \in \mathbb{F}_{p^2}$. We conclude that $s = ur \in \mathbb{F}_{p^n}$, with $\deg r = n-2$ and $u \in \mathbb{F}_{p^2}$. □

Now we apply the technique described in Sec. 4.1 to reduce the coefficient size of r in the JLSV$_1$ construction. We have $r_{n-1} = 0$ and we assume that $r_{n-2} = 1$. We define the lattice by the $(n-1) \times (n-1)$ matrix

$$L = \underbrace{\begin{bmatrix} p & & & \\ & \ddots & & \\ & & p & \\ \boldsymbol{r}_0 & \dots & \boldsymbol{r}_{n-3} & 1 \end{bmatrix}}_{n-1 \times n-1} \begin{matrix} 0 \\ \vdots \\ n-3 \\ n-2 \end{matrix} \begin{matrix} \Big\} \\ \Big\} \\ \Big\} \\ \Big\} \end{matrix} \begin{matrix} n-2 \text{ rows} \\ \\ \\ \text{row } n-2 \text{ with } \boldsymbol{r} \text{ coeffs} \end{matrix}$$

After reducing the lattice with LLL, we obtain an element $\boldsymbol{r}'$ whose coefficients are bounded by $Cp^{\frac{n-2}{n-1}}$. The norm of $\boldsymbol{r}'$ in the number field K_f constructed with the JLSV$_1$ method is

$$\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}') = O(p^{\frac{3}{2}n-2-\frac{1}{n-1}}) = O(Q^{\frac{3}{2}-\frac{2}{n}-\frac{1}{n(n-1)}}) \ .$$

This is better than the previous $O\big(Q^{\frac{3}{2}-\frac{3}{2n}}\big)$ case: the norm is smaller by a factor of size $O\big(Q^{\frac{1}{2}n+\frac{1}{n(n-1)}}\big)$. For $n=4$, we obtain $\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}') = O\big(Q^{\frac{11}{12}}\big)$, which is strictly less than $O(Q)$.

We can do even better by re-using the element $\boldsymbol{r}$ of degree $n-2$ and the given one s of degree $n-1$, and combining them.

Generalization to subfields of higher degrees. It was pointed out to us by an anonymous reviewer that more generally, by standard linear algebra arguments, for $m \mid n$ and $s \in \mathbb{F}_{p^n}$, there exists a non-zero $u \in \mathbb{F}_{p^m}$ such that $s \cdot u$ is a polynomial of degree at most $n-m$.

5.2 Smaller Preimage Norm

First, suppose that the target element $s = \sum_{i=0}^{n-1} s_i x^i$ satisfies $s_{n-1} = 0$ and $s_{n-2} = 1$. We can define a lattice whose vectors, once mapped to $\mathbb{F}_{p^n}$, are either 0 (so vectors are sums of multiples of p and ψ) or are multiples of the initial target s, satisfying Lemma 2. The above r of degree $n-2$ is a good candidate. The initial s also. If there is no initial s of degree $n-1$, then simply take at random any u in a proper subfield of $\mathbb{F}_{p^n}$ which is not $\mathbb{F}_p$ itself and set $s = u \cdot r$. Then s will have $s_{n-1} \neq 0$. Then define the lattice

$$L = \begin{bmatrix} p & & & & \\ & \ddots & & & \\ & & p & & \\ \boldsymbol{r}_0 & \dots & \boldsymbol{r}_{n-3} & 1 & \\ \boldsymbol{s}_0 & \dots & \boldsymbol{s}_{n-3} & \boldsymbol{s}_{n-2} & 1 \end{bmatrix}_{n\times n} \begin{matrix} 0 \\ \vdots \\ n-3 \\ n-2 \\ n-1 \end{matrix} \begin{matrix} \Big\} \\ \Big\} \\ \Big\} \\ \Big\} \\ \Big\} \end{matrix} \begin{matrix} n-2 \text{ rows} \\ \\ \\ \text{row } n-2 \text{ with } \boldsymbol{r} \text{ coeffs} \\ \text{row } n-1 \text{ with } \boldsymbol{s} \text{ coeffs} \end{matrix}$$

and use it in place of the lattices of Sec. 4.1 or 4.2.

5.3 Summary of results

We give in Table 3 the previous and new upper bounds for the norm of s in a number field K_f for three polynomial selection methods: the JLSV_1 method, the generalized Joux–Lercier method and the Conjugation method, and the complexity of the booting step to find a B-smooth decomposition of $\mathrm{Norm}_{K_f/\mathbb{Q}}(s)$. We give our practical results for small n, where there are the most dramatic improvements. We obtain the optimal norm size of $Q^{\varphi(n)/n}$ for $n = 2, 3, 5$ with the gJL method and also for $n = 4$ with the Conjugation method.

6 Practical examples

We present an example for each of the three polynomial selection methods we decided to study. The Conjugation method provides the best timings for $\mathbb{F}_{p^2}$ at 180 dd [6]. We apply the gJL method to $\mathbb{F}_{p^3}$ according to [6, Fig. 3]. We decided to use the JLSV_1 method for $\mathbb{F}_{p^4}$ [6, Fig. 4].

Table 3. Norm bound of the preimage with our method, and booting step complexity.

$\mathbb{F}_{p^n}$	poly. selec.	norm bound			booting step $L_Q[\frac{1}{3}, c]$		practical values of c				
		nothing	JLSV	this work	prev	this work	$n=2$	$n=3$	$n=4$	$n=5$	$n=6$
any $n>1$	gJL	$Q^{1+\frac{1}{n}}$	Q	$Q^{1-1/n}$	1.44	$(3(1-\frac{1}{n}))^{1/3}$	1.14	1.26	–	1.34	–
even $n \geq 4$				$Q^{1-2/n}$		$(3(1-\frac{2}{n}))^{1/3}$	–	–	1.14	–	1.26
any $n>1$	Conj	Q^2	Q	$Q^{1-1/n}$	1.44	$(3(1-\frac{1}{n}))^{1/3}$	1.14	1.26	–	1.34	–
even $n \geq 4$				$Q^{1-2/n}$		$(3(1-\frac{2}{n}))^{1/3}$	–	–	1.14	–	1.26
any $n>1$	JLSV_1	$Q^{\frac{3}{2}-\frac{1}{2n}}$	Q^2	$Q^{3/2-3/(2n)}$	1.65	$(\frac{9}{2}(1-\frac{1}{n}))^{1/3}$	1.31	1.44	–	1.53	–
even $n \geq 4$				$Q^{3/2-5/(2n)}$		$(\frac{3}{2}(3-\frac{5}{n}))^{1/3}$	–	–	1.38	–	1.48

6.1 Examples for Small n and p^n of 180 Decimal Digits (dd)

Example for $n = 2$, Conjugation Method. We take the parameters of the record in [6]: p is a 90 decimal digit (300 bit) prime number, and f, ψ are computed with the Conjugation method. We choose a target s from the decimal digits of exp(1).

$$
\begin{aligned}
p &= 314159265358979323846264338327950288419716939937510582097494459230781640628620899877709223\\
f &= x^4 + 1\\
\psi &= x^2 + 10778151309582301866698988310224439480941229764389534909741063250804945537669878469169959 3\ x + 1\\
s &= 271828182845904523536028747135319858432320810108854154561922281807332337576949857498874314\ x\\
&\quad + 958880662507673263211420165757531990227722354115265486848084409739492084711947246180906 92
\end{aligned}
$$

We first compute $s' = \frac{1}{s_0} s$ then reduce

$$
L = \begin{bmatrix} p & 0 & 0 & 0 \\ s'_0 & 1 & 0 & 0 \\ 1 & \psi_1 & 1 & 0 \\ 0 & 1 & \psi_1 & 1 \end{bmatrix}
$$

then $\text{LLL}(L)$ produces $\boldsymbol{r}$ of degree 3 and coefficient size $O(p^{1/4})$. Actually LLL outputs four short vectors, hence we get four small candidates for $\boldsymbol{r}$, each of norm $\text{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = O(p) = O(Q^{1/2}) = O(Q^{\varphi(n)/n})$, i.e. 90 dd. To slightly improve the smoothness search time, we can compute linear combinations of these four reduced preimages.

$$
\begin{aligned}
&3603397286457205828471 x^3 + 13679035553643009711078 x^2 + 557746247085194895659 4 x + 856176942703613067714\\
&9219461324482190814893 x^3 - 4498175796333854926013 x^2 + 895775002549467382219 8 x + 111788824169113006040 9\\
&2826839094462418314170 2 x^3 + 5699666741226225385259 x^2 - 178019404032168663329 11 x + 544843224771048269684 8\\
&3352162792941463140060 x^3 + 3212585012235692902287 x^2 - 557063651808475912551 3 x + 469265082905446625423 27
\end{aligned}
$$

The norm of the first element is

$$
\text{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = 213988280295201686111690452803024284348669666570970757613375980707604853409486778001629 21
$$

of 90 decimal digits, as expected. For a close to optimal running-time of $L_Q[1/3, 1.14] \sim 2^{40}$ to find a boot, the special-q bound would be around 64 bits.

Example for $n = 3$, gJL Method. We take p of 60 dd (200 bits) so that $\mathbb{F}_{p^3}$ has size 180 dd (600 bits) as above. We took p a prime made of the 60 first decimal digits of π. We constructed f, ψ, g with the gJL method described in [6].

$$
\begin{aligned}
p &= 314159265358979323846264338327950288419716939937510582723487\\
f &= x^4 - x + 1\\
\psi &= x^3 + 227138144243642333129902287795664772043667053260089299478579x^2\\
&\quad + 126798022201426805402186761110440110121157863791585328913565x + 863983091574414435397918995177883881848539630718471155526 38\\
g &= 287767088987135456608033317246385224990821439 1x^3 + 609951652432557506082184162014047061886340388 1x^2\\
&\quad - 101235332348344733160532896231657564372672984 03x + 202907337179191496597604128420820845026712 0556\\
s &= 271828182845904523536028747135319858432320810108854154561922x^2 + 281807332337576949857498874314095888066250767326321142016575x\\
&\quad + 753199022772235411526548684808589516264937392972591398598 75
\end{aligned}
$$

We set $s' = \frac{1}{s_2}s$. The lattice to be reduced is

$$L = \begin{bmatrix} p & 0 & 0 & 0 \\ 0 & p & 0 & 0 \\ s'_0 & s'_1 & 1 & 0 \\ \psi_0 & \psi_1 & \psi_2 & 1 \end{bmatrix}$$

then LLL(L) computes four short vectors $\boldsymbol{r}$ of degree 3, of coefficient size $O(p^{1/2})$, and of norm size $\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = O(p^2) = O(Q^{2/3}) = O(Q^{\varphi(n)/n})$.

$$
\begin{aligned}
&1597749306375059000939093070 18x^3 + 165819631832105094449987774814x^2 + 177828199322419553601266354904x - 159912786936943488400590389195\\
&1365830293545209052324129410 48x^3 - 521269847225531188433352927453x^2 + 322722415562853671586868492721x + 255238068915917937217884608875\\
&1182890075989340687266630002 66x^3 + 499013489972894059858543976363x^2 - 105084220861844155797015713666x + 535978811382585906107397024241\\
&4116038900545395001314743137 73x^3 - 240161030577722451131067159670x^2 - 373289346204280810310169575030x - 389720783049275894296185820094
\end{aligned}
$$

The norm of the first element is

$$\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r}) = 997840136509677868374734441582077227769466501519927620849763845265357390584602475858356409809239812991892769866071779$$

of 117 decimal digits (with $\frac{2}{3}180 = 120$ dd). For a close to optimal running-time of $L_Q[1/3, 1.26] \sim 2^{45}$ to find a boot, the special-q bound would be around 77 bits.

Example for $n = 4$, JLSV$_1$ Method.

$$
\begin{aligned}
p &= 31415926535897932384626433832795028841998001 1\\
\ell &= 49348022005446793094172454999380755676651143247932834802731698819521755649884772819780061\\
f = \psi &= x^4 + x^3 + 708981540362206410931 62x^2 + x + 1\\
g &= 10191609642706717156787 2x^4 + 10191609642706717156787 2x^3 + 22080632887404989855101 1x^2\\
&\quad + 10191609642706717156787 2x + 1019160964270671715678 72\\
s &= 271828182845904523536028747135319858432320810x^3 + 108854154561922281807332337576949857498874314x^2\\
&\quad + 95888066250767326321142016575753199022772235x + 41152654868480844097394920847127588391952018
\end{aligned}
$$

We set $s' = \frac{1}{s_3}s$. The subfield simplification for s gives

$$r = x^2 + 1349691223972631029797432269152823554001619 11x + 1046424406499377563685457653347410492071210 11 .$$

We reduce the lattice defined by

$$L = \begin{bmatrix} p & 0 & 0 & 0 \\ 0 & p & 0 & 0 \\ r_0 & r_1 & 1 & 0 \\ s'_0 & s'_1 & s'_2 & 1 \end{bmatrix}$$

then LLL(L) produces these four short vectors of degree 3, coefficient size $O(p^{1/2})$, and norm $\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r'}) = O(p^{\frac{7}{2}}) = O(Q^{7/8})$ (smaller than $O(Q)$).

$$5842961997149263751946x^3 + 2907368273308610113 76x^2 - 5618779793817086743792x + 1092949800287557029045$$
$$1640842643903161175359x^3 + 1555259026913188958957 5x^2 - 4425488394163838271378x - 5734086421794811858814$$
$$6450686906504525374853x^3 + 1376877124265095739941 9x^2 + 1061758394423409088057 9x + 16261617079167797580912$$
$$1692913580413987886539 1x^3 + 6981855717048102583 44x^2 + 1279930041102246114079x - 2278728269871806528 4157$$

The norm of the first element is

$$\mathrm{Norm}_{K_f/\mathbb{Q}}(\boldsymbol{r'}) = 1452143929217271115166861110413357998278729994931024260194421897764500704952 7\backslash$$
$$0123656021783074136945302749067576757516984664647990043605467452102146421782 85$$

of 155 decimal digits (with $\frac{7}{8}180 = 157.5$). For a close to optimal running-time of $L_Q[1/3, 1.34] \sim 2^{49}$ to find a boot, the special-q bound would be approximately of 92 bits. This is very large however.

6.2 Experiments: finding boots for $\mathbb{F}_{p^4}$ of 120 dd

We experimented our booting step method for $\mathbb{F}_{p^4}$ of 120 dd (400 bits). Without the quadratic subfield simplification, the randomized target norm is bounded by $Q^{9/8}$ of 135 dd (450 bits). The largest special-q in the boot has size $L_Q[2/3, 3/4]$ (25 dd, 82 bits) according to Lemma 1 with $e = 9/8$. The running-time to find one boot would be $L_Q[1/3, 1.5] \sim 2^{44}$.

We apply the quadratic subfield simplification. The norm of the randomized target is $Q^{7/8}$ of 105 dd ($\simeq$ 350 bits). We apply theorem 1 with $e = 7/8$. The size of the largest special-q in the boot will be approximately $L_Q[2/3, 0.634]$ which is 21 dd (69 bits). The running-time needed to find one boot with the special-q of no more than 21 dd is $L_Q[1/3, 1.38] \sim 2^{40}$ (to be compared with the dominating part of NFS-DL of $L_Q[1/3, 1.923] \sim 2^{57}$). We wrote a magma program to find boots, using GMP-ECM for q-smooth tests. We first set a special-q bound of 70 bits and obtained boots in about two CPU hours. We then reduced the special-q bound to a machine word size (64 bits) and also found boots in around two CPU hours. We used an Intel Xeon E5-2609 0 at 2.40GHz with 8 cores.

7 Conclusion

We have presented a method to improve the booting step of individual logarithm computation, the final phase of the NFS algorithm. Our method is very efficient for small n, combined with the gJL or Conjugation methods; it is also usefull for the JLSV_1 method, but with a slower running-time. For the moment, the booting step remains the dominating part of the final individual discrete logarithm. If our method is improved, then special-q descent might become the new bottleneck in some cases. A lot of work remains to be done on final individual logarithm computations in order to be able to compute one individual logarithm as fast as was done in the Logjam [2] attack, especially for $n \geq 3$.

Acknowledgements. The author thanks the anonymous reviewers for their constructive comments and the generalization of Lemma 3. The author is grateful to Pierrick Gaudry, François Morain and Ben Smith.

References

1. Adleman, L.: A subexponential algorithm for the discrete logarithm problem with applications to cryptography. In: 20th FOCS. pp. 55–60. IEEE Computer Society Press (Oct 1979)
2. Adrian, D., Bhargavan, K., Durumeric, Z., Gaudry, P., Green, M., Halderman, J.A., Heninger, N., Springall, D., Thomé, E., Valenta, L., VanderSloot, B., Wustrow, E., Béguelin, S.Z., Zimmermann, P.: Imperfect forward secrecy: How Diffie-Hellman fails in practice. In: Ray, I., Li, N., Kruegel:, C. (eds.) ACM CCS 15. pp. 5–17. ACM Press (Oct 2015)
3. Barbulescu, R., Pierrot, C.: The multiple number field sieve for medium- and high-characteristic finite fields. LMS J. Comput. Math. 17, 230–246 (1 2014), http://journals.cambridge.org/article_S1461157014000369
4. Barbulescu, R.: Algorithmes de logarithmes discrets dans les corps finis. Ph.D. thesis, Université de Lorraine (2013), https://tel.archives-ouvertes.fr/tel-00925228
5. Barbulescu, R., Gaudry, P., Guillevic, A., Morain, F.: Discrete logarithms in $\text{GF}(p^2)$ — 180 digits (2014), https://listserv.nodak.edu/cgi-bin/wa.exe?A2=NMBRTHRY;2ddabd4c.1406, announcement available at the NMBRTHRY archives
6. Barbulescu, R., Gaudry, P., Guillevic, A., Morain, F.: Improving NFS for the discrete logarithm problem in non-prime finite fields. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 129–155. Springer, Heidelberg (Apr 2015), http://hal.inria.fr/hal-01112879
7. Barbulescu, R., Gaudry, P., Joux, A., Thomé, E.: A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 1–16. Springer, Heidelberg (May 2014)
8. Blake, I.F., Mullin, R.C., Vanstone, S.A.: Computing logarithms in $\text{GF}(2^n)$. In: Blakley, G.R., Chaum, D. (eds.) CRYPTO'84. LNCS, vol. 196, pp. 73–82. Springer, Heidelberg (Aug 1984)

9. Canfield, E.R., Erdös, P., Pomerance, C.: On a problem of Oppenheim concerning "factorisatio numerorum". J. Number Theory 17(1), 1–28 (1983)
10. Chen, Y.: Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe. Ph.D. thesis, Université Paris 7 Denis Diderot (2013), http://www.di.ens.fr/~ychen/research/these.pdf
11. Commeine, A., Semaev, I.: An algorithm to solve the discrete logarithm problem with the number field sieve. In: Yung, M., Dodis, Y., Kiayias, A., Malkin, T. (eds.) PKC 2006. LNCS, vol. 3958, pp. 174–190. Springer, Heidelberg (Apr 2006)
12. Coppersmith, D.: Modifications to the number field sieve. Journal of Cryptology 6(3), 169–180 (1993)
13. Coppersmith, D., Odlzyko, A.M., Schroeppel, R.: Discrete logarithms in GF(p). Algorithmica 1(1-4), 1–15 (1986), http://dx.doi.org/10.1007/BF01840433
14. Gama, N., Nguyen, P.Q.: Predicting lattice reduction. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 31–51. Springer, Heidelberg (Apr 2008)
15. Gordon, D.M.: Discrete logarithms in GF(p) using the number field sieve. SIAM J. Discrete Math 6, 124–138 (1993)
16. Hayasaka, K., Aoki, K., Kobayashi, T., Takagi, T.: An experiment of number field sieve for discrete logarithm problem over GF(p^{12}). In: Fischlin, M., Katzenbeisser, S. (eds.) Number Theory and Cryptography, LNCS, vol. 8260, pp. 108–120. Springer (2013), http://dx.doi.org/10.1007/978-3-642-42001-6_8
17. Joux, A., Lercier, R.: Improvements to the general number field for discrete logarithms in prime fields. Math. Comp. 72(242), 953–967 (2003)
18. Joux, A., Lercier, R., Naccache, D., Thomé, E.: Oracle-assisted static Diffie-Hellman is easier than discrete logarithms. In: Parker, M.G. (ed.) 12th IMA International Conference on Cryptography and Coding. LNCS, vol. 5921, pp. 351–367. Springer, Heidelberg (Dec 2009)
19. Joux, A., Lercier, R., Smart, N., Vercauteren, F.: The number field sieve in the medium prime case. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 326–344. Springer, Heidelberg (Aug 2006)
20. Kalkbrener, M.: An upper bound on the number of monomials in determinants of sparse matrices with symbolic entries. Mathematica Pannonica 73, 82 (1997)
21. Lenstra, A., Lenstra, H.W., J., Lovász, L.: Factoring polynomials with rational coefficients. Mathematische Annalen 261(4), 515–534 (1982), http://dx.doi.org/10.1007/BF01457454
22. Matyukhin, D.: Effective version of the number field sieve for discrete logarithms in the field GF(p^k) (in Russian). Trudy po Discretnoi Matematike 9, 121–151 (2006), http://m.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=tdm&paperid=144&option_lang=eng
23. Pierrot, C.: The multiple number field sieve with conjugation and generalized joux-lercier methods. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 156–170. Springer, Heidelberg (Apr 2015)
24. Weber, D.: Computing discrete logarithms with quadratic number rings. In: Nyberg, K. (ed.) EUROCRYPT'98. LNCS, vol. 1403, pp. 171–183. Springer, Heidelberg (May / Jun 1998)
25. Zajac, P.: Discrete Logarithm Problem in Degree Six Finite Fields. Ph.D. thesis, Slovak University of Technology (2008), http://www.kaivt.elf.stuba.sk/kaivt/Vyskum/XTRDL