Traps to the BGJT-Algorithm for Discrete Logarithms

Qi Cheng
School of Computer Science
University of Oklahoma
Norman, OK
qcheng@cs.ou.edu

Daqing Wan
Department of Mathematics
University of California
Irvine, CA
dwan@math.uci.edu

Jincheng Zhuang
University of Oklahoma
Norman, OK
jzhuang@ou.edu

Abstract

In the recent breakthrough paper by Barbulescu, Gaudry, Joux and Thomé, a quasi-polynomial time algorithm (QPA) is proposed for the discrete logarithm problem over finite fields of small characteristic. The time complexity analysis of the algorithm is based on several heuristics presented in their paper. We show that some of the heuristics are problematic in their original forms, in particular, when the field is not a Kummer extension. We believe that the basic idea behind the new approach should still work, and propose a fix to the algorithm in non-Kummer cases, without altering the quasi-polynomial time complexity. The modified algorithm is also heuristic. Further study is required in order to fully understand the effectiveness of the new approach.

1 Introduction

Many cryptography protocols rely on hard computational number theoretical problems for security. The discrete logarithm problem over finite fields is one of the most important candidates, besides the integer factorization problem. The hardness of discrete logarithms underpins the security of the widely adopted Diffie-Hellman key exchange protocol [5] and ElGamal's cryptosystem [6].

The state-of-the-art general-purpose methods for solving the discrete logarithm problem in finite fields are the number field sieve and the function field sieve, which originated from the index-calculus algorithm. All the algorithms run in subexponential time. Let

$$L_N(\alpha) = \exp(O((\log N)^{\alpha}(\log\log N)^{1-\alpha})).$$

For a finite field $\mathbf{F}_q$, successful efforts have been made to reduce the heuristic complexity of these algorithms from $L_q(1/2)$ to $L_q(1/3)$. See [16, 1, 14, 4, 8, 2, 12, 13].

A sequence of breakthrough results [10, 11, 7] recently on the discrete logarithm problem over finite fields culminated in a discovery of a quasi-polynomial algorithm for small characteristic fields [3]. For a finite field $\mathbf{F}_{q^{2k}}$ with $k < q$, their algorithm runs in heuristic time $q^{O(\log k)}$. This result, if correct, essentially removes the discrete logarithm over small characteristic fields from hard problems in cryptography.

1.1 Where does the computation really happen?

Most serious attacks on the discrete logarithm problem over finite fields are based on smoothness of integers or polynomials. A polynomial is m-smooth if all its irreducible factors have degrees $\leq m$. The probability that a random polynomial of degree n ($\geq m$) over a finite field $\mathbf{F}_q$ is $m-$smooth is about $(n/m)^{-n/m}$ [15].

Suppose that we need to compute discrete logarithm in the field $\mathbf{F}_{q^{2k}}$ where $q > k > 1$. A main technique in [3], which bases on smooth polynomials, is to find a nice ring generator ζ of $\mathbf{F}_{q^{2k}} = \mathbf{F}_{q^2}[\zeta]$ over $\mathbf{F}_{q^2}$ satisfying

$$x^q = h_0(x)/h_1(x),$$

where h_1 and h_0 are polynomials of very small degree. In many places of the computation, polynomial degrees can be dropped quickly by replacing x^q with $h_0(x)/h_1(x)$, which allows an effective attack based on smoothness.

The main issue with this approach is that the computation really takes place in the ring $\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x))$, where in the analysis of [3], the computation is assumed to be in $\mathbf{F}_{q^2}[x]/(f(x))$, where $f(x)$ is the minimal polynomial of ζ over $\mathbf{F}_{q^2}$. Since $f(x)$ divides $x^q h_1(x) - h_0(x)$, there is a natural surjective ring homomorphism

$$\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x)) \to \mathbf{F}_{q^2}[x]/(f(x)).$$

But the former ring, which is a direct sum of the latter field (if $f(x)$ is a simple factor of $x^q h_1(x) - h_0(x)$) and a few other rings, is much larger in many cases. The computation thus can be affected by the other rings, rendering several conjectures in [11, 3] problematic.

1.2 Our work

Interestingly, for the Kummer extension of the form $\mathbf{F}_{q^2}[x]/(x^{q-1} - a)$, everything is fine. This is because the difference between the ring $\mathbf{F}_{q^2}[x]/(x^q - ax)$ and the

field is rather small. The discrete logarithm of x, which is a zero divisor in the former ring, can be computed easily in the latter field, since it belongs to a subgroup of a small order (dividing $(q-1)(q^2-1)$) in the field. This is consistent with all announced practical implementations.

However, in case of more difficult non-Kummer extensions, we discover that there are multiple problems. First, if $x^q h_1(x) - h_0(x)$ has linear factors over $\mathbf{F}_{q^2}$, the discrete logarithms of these linear factors cannot be computed in polynomial time, invalidating a basic assumption in [3]. One can verify that most of polynomials given in [11, Table 1] have linear factors. Second, even at the stage of finding discrete logarithms of linear elements, we show that there are additional serious restrictions on the choice of h_0 and h_1. For example, if $x^q h_1(x) - h_0(x)$ has another irreducible factor over $\mathbf{F}_{q^2}$ of degree k_i satisfying $\gcd(k_i, k) > 1$, we do not see how the algorithm can work. We propose to select h_0 and h_1 such that $x^q h_1(x) - h_0(x)$ has only one irreducible factor $f(x)$ over $\mathbf{F}_{q^2}$ of degree k, and all other irreducible factors over $\mathbf{F}_{q^2}$ have degrees bigger than one and relatively prime to k. Under these assumptions, we give an algorithm which will find the discrete logarithm of any linear element in polynomial time, under a heuristic assumption supported by our theoretical results and numerical data.

For a non-linear element, a clever idea, the so-called QPA-descent, was proposed in [3] to reduce its degree, until its relation to linear factors can be found. While the above two problems about linear factors can be fixed under our newly improved heuristic assumptions, another serious problem is that there are *traps* in the QPA-descent. For these traps, the QPA-descent described in [3] will not work at all. They will also block the descent of other elements, hence severely affecting the usefulness of the new algorithm. We propose a descent strategy that avoids the traps, without altering the quasi-polynomial time complexity. The modified algorithm is also heuristic. We have done a few numerical studies to confirm the heuristic.

In summary, for large non-Kummer fields, we believe that the problem can be significantly more subtle than previously thought and further study needs to be conducted in order to fully understand the effect of the new algorithm.

2 Finding the discrete logarithm of the linear factors

We first review the new algorithm in [3]. Suppose that the discrete logarithm is sought over the field $\mathbf{F}_{q^{2k}}$ with $k < q$. For other small characteristic fields, for example, $\mathbf{F}_{p^k}$ ($p < k$), one first embeds it into a slightly larger field:

$$\mathbf{F}_{p^k} \to \mathbf{F}_{q^k} \to \mathbf{F}_{q^{2k}}$$

where $q = p^{\lceil \log_p k \rceil}$. A quasi-polynomial time algorithm for $\mathbf{F}_{q^{2k}}$ implies a quasi-polynomial time algorithm for $\mathbf{F}_{p^k}$. We assume that

$$\mathbf{F}_{q^{2k}} = \mathbf{F}_{q^2}[\zeta]$$

where $\zeta^q = \frac{h_0(\zeta)}{h_1(\zeta)}$. Here h_0 and h_1 are polynomials over $\mathbf{F}_{q^2}$ relatively prime to each other, and of a constant degree. In particular, $\deg(h_0) < q + \deg(h_1)$. To find such a nice ring generator ζ, one searches over all the polynomials $h_0(x)$ and $h_1(x)$ of a constant degree in $\mathbf{F}_{q^2}[x]$, until $h_1(x)x^q - h_0(x)$ has an irreducible factor $f(x)$ of degree k with multiplicity one. Let the factorization be

$$x^q h_1(x) - h_0(x) = f(x) \prod_{i=1}^{l} (f_i(x))^{a_i} \tag{1}$$

where the polynomials $f(x)$ and $f_i(x)$'s are irreducible and pair-wise prime. Denote the degree of $f_i(x)$ by k_i.

Remark 1 *In practice, it is enough to search only a quadratic polynomial h_0 (not necessarily monic) and a monic linear polynomial h_1 in $\mathbf{F}_{q^2}[x]$. However proving the existence of such polynomials for any constant degree such that $x^q h_1(x) - h_0(x)$ has the desired factorization pattern seems to be out of reach by current techniques.*

For simplicity we assume that $h_1(x)$ is monic and linear. Most of the known algorithms start by computing the discrete logarithms of elements in a special set called a factor base, which usually contains small integers, or low degree polynomials. In the new approach [11, 3], the factor base consists of the linear polynomials $\zeta + \alpha$ for all $\alpha \in \mathbf{F}_{q^2}$, and an algorithm is designed to compute the discrete logarithms of all the elements in the factor base. It is conjectured that this algorithm runs in polynomial time. One starts the algorithm with the identity:

$$\prod_{\alpha \in \mathbf{F}_q} (x - \alpha) = x^q - x.$$

Then apply the Mobius transformation

$$x \mapsto \frac{ax + b}{cx + d}$$

where the matrix $m = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathbf{F}_{q^2}^{2\times 2}$ is nonsingular. We have

$$\prod_{\alpha \in \mathbf{F}_q} (\frac{ax + b}{cx + d} - \alpha) = (\frac{ax + b}{cx + d})^q - \frac{ax + b}{cx + d}$$

Clearing the denominator:

$$\begin{aligned}
& (cx+d)\prod_{\alpha\in\mathbf{F}_q}((ax+b)-\alpha(cx+d)) \\
= \;& (ax+b)^q(cx+d)-(ax+b)(cx+d)^q \\
= \;& (a^qx^q+b^q)(cx+d)-(ax+b)(c^qx^q+d^q).
\end{aligned}$$

Multiplying both sides by $h_1(x)$ and replacing $x^qh_1(x)$ by $h_0(x)$, we obtain

$$\begin{aligned}
& h_1(x)(cx+d)\prod_{\alpha\in\mathbf{F}_q}((ax+b)-\alpha(cx+d)) \\
= \;& (a^qh_0(x)+b^qh_1(x))(cx+d)-(ax+b)(c^qh_0(x)+d^qh_1(x)) \\
& \pmod{x^qh_1(x)-h_0(x)}.
\end{aligned}$$

If the right-hand side can be factored into a product of linear factors over $\mathbf{F}_{q^2}$, we obtain a relation of the form

$$\lambda^{e_0}\prod_{i=1}^{q^2}(x+\alpha_i)^{e_i}=\prod_{i=1}^{q^2}(x+\alpha_i)^{e_i'} \pmod{x^qh_1(x)-h_0(x)}, \tag{2}$$

where λ is a multiplicative generator of $\mathbf{F}_{q^2}$, $\alpha_1=0,\alpha_2,\alpha_3,\ldots,\alpha_{q^2}$ is a natural ordering of elements in $\mathbf{F}_{q^2}$, and e_i's and e_i''s are non-negative integers.

Following the same notations in [3], let $\mathcal{P}_q$ be a set of representatives of the left cosets of $PGL_2(\mathbf{F}_q)$ in $PGL_2(\mathbf{F}_{q^2})$. Note that the cardinality of $\mathcal{P}_q$ is q^3+q. It was shown in [3] that the matrices in the same coset produce the same relation (2).

Suppose that for some $1\le g\le q^2$, $\zeta+\alpha_g$ is a known multiplicative generator of $\mathbf{F}_{q^2}[\zeta]=\mathbf{F}_{q^2}[x]/(f(x))$. Since (2) also holds modulo $f(x)$, taking the discrete logarithm w.r.t. the base $\zeta+\alpha_g$, we obtain

$$e_0\log_{\zeta+\alpha_g}\lambda+\sum_{1\le i\le q^2,i\ne g}(e_i-e_i')\log_{\zeta+\alpha_g}(\zeta+\alpha_i)\equiv e_g'-e_g \pmod{q^{2k}-1}. \tag{3}$$

The above equation gives us a linear relation among the discrete logarithm of linear factors. One hopes to collect enough relations such that the linear system formed by those relations is non-singular over $\mathbf{Z}/(q^{2k}-1)\mathbf{Z}$. It allows us to solve $\log_{\zeta+\alpha_g}(\zeta+\alpha_i)$ for all the $\zeta+\alpha_i$ in the factor base.

However, if for some $1\le z\le q^2$,

$$(x+\alpha_z)|x^qh_1(x)-h_0(x),$$

the algorithm will unlikely compute $\log_{\zeta+\alpha_g}(\zeta+\alpha_z)$. It is because that $x+\alpha_z$ is zero or nilpotent (w.l.o.g. let $f_1 = x+\alpha_z$) in the $\mathbf{F}_{q^2}[x]/((x+\alpha_z)^{a_1})$ component of the ring

$$\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x)) = \mathbf{F}_{q^2}[x]/(f(x)) \oplus \bigoplus_{i=1}^{l} \mathbf{F}_{q^2}[x]/(f_i(x)^{a_i}).$$

Hence in (2), if $e_z > 0$, e'_z is positive as well. Most likely we will have $e_z = e'_z$, so the coefficient for $\log_{\zeta+\alpha_g}(\zeta+\alpha_z)$ in (3) will always be 0.

Remark 2 *If $e'_z > e_z \geq 1$, it is possible to compute $\log_{\zeta+\alpha_g}(\zeta+\alpha_z)$. However, this requires the low degree polynomial in the right hand side of (2) to have the factor $(x+\alpha_z)^2$, which is unlikely. Our numerical data confirm that it never happens when q is sufficiently large.*

To compute the discrete logarithm of $\zeta+\alpha_z$, we have to use additional relations which hold for the field $\mathbf{F}_{q^2}[\zeta]$ but may not hold for the bigger ring $\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x))$. The equation

$$(\zeta+\alpha_z)^{q^{2k}-1} = 1$$

is such an example. But this does not help in computing its discrete logarithm in the field $\mathbf{F}_{q^2}[\zeta]$, if it is the only relation involving $\zeta+\alpha_z$.

In general, it is hard to find useful additional relations for $x+\alpha_z$, since for the algorithm to work, it is essential that we replace x^q by $h_0(x)/h_1(x)$ (not replace $f(x)$ by zero) in the relation generating stage. Hence it is not clear that the discrete logarithm of $\zeta+\alpha_z$ can be computed in polynomial time, invalidating a conjecture in [3].

Remark 3 *An exception is in the case of a Kummer extension, where the zero divisor x in the ring has a small order in the field.*

3 The tale of two lattices

To fix the above problem in a non-Kummer case, we can either change our factor base to not include the linear factors of $x^q h_1(x) - h_0(x)$, or we can search for h_0 and h_1 such that $x^q h_1(x) - h_0(x)$ does not have linear factors. In the following discussion, we will assume that $x^q h_1(x) - h_0(x)$ has no linear factor for simplicity. That is,

$$k_i := \deg(f_i) \geq 2 \ (1 \leq i \leq l).$$

In this case, the linear factors $x+\alpha_i$'s are invertible in the ring $\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x))$ and equation (2) reduces to

$$\lambda^{e_0} \prod_{i=1}^{q^2} (x+\alpha_i)^{e_i - e_i'} = 1 \pmod{x^q h_1(x) - h_0(x)}. \tag{4}$$

We define two fundamental lattices in $\mathbf{Z}^{q^2+1}$:

$$\begin{aligned} \mathcal{L}_1 &= \{(e_0, e_1, \ldots, e_{q^2}) | \lambda^{e_0} \prod_{i=1}^{q^2} (x+\alpha_i)^{e_i} = 1 \pmod{f(x)}\}, \\ \mathcal{L}_2 &= \{(e_0, e_1, \ldots, e_{q^2}) | \lambda^{e_0} \prod_{i=1}^{q^2} (x+\alpha_1)^{e_i} = 1 \pmod{x^q h_1(x) - h_0(x)}\}. \end{aligned}$$

It is easy to see that $\mathcal{L}_2 \subseteq \mathcal{L}_1$. Consider the group homomorphism

$$\psi_1 : \mathbf{Z}^{q^2+1} \to (\mathbf{F}_{q^2}[x]/(f(x)))^*$$

given by

$$(e_0, e_1, \ldots, e_{q^2}) \mapsto \lambda^{e_0} \prod_{i=1}^{q^2} (x+\alpha_i)^{e_i}.$$

The group homomorphism ψ_2 is defined in the same way, except that modulo $f(x)$ is replaced by modulo $(x^q h_1(x) - h_0(x))$ respectively.

Theorem 1 *If* $\deg(h_1) \leq 2$*, then the maps* ψ_1 *and* ψ_2 *are surjective.*

Proof: It is enough to prove that ψ_2 is surjective. If not, the image H of ψ_2 would be a proper subgroup of $(\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x)))^*$. We can then choose a non-trivial character χ of $(\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x)))^*$ which is trivial on the subgroup H. Since χ is trivial on H which contains $\mathbf{F}_{q^2}^*$, we can use the Weil bound as given in Theorem 2.1 in [17] and deduce that

$$1 + q^2 = |1 + \sum_{\alpha \in \mathbf{F}_{q^2}} \chi(x+\alpha)| \leq (q + \deg(h_1) - 2)\sqrt{q^2} \leq q^2.$$

This is a contradiction. It follows that ψ_2 must be surjective. □

Note that in the application of computing discrete logarithms, it is important that ψ_1 is surjective. As a corollary, we deduce

Corollary 1 *If* $\deg(h_1) \leq 2$*, then*

- *the group* $\mathbf{Z}^{q^2+1}/\mathcal{L}_1$ *is isomorphic to the cyclic group* $\mathbf{Z}/(q^{2k}-1)\mathbf{Z}$.
- *the group* $\mathbf{Z}^{q^2+1}/\mathcal{L}_2$ *is isomorphic to*

$$\mathbf{Z}/(q^{2k}-1)\mathbf{Z} \oplus \bigoplus_{i=1}^{l} \mathbf{Z}/(q^{2k_i}-1)\mathbf{Z} \bigoplus (a \textit{ finite } p\textit{-group}).$$

In particular, the group $\mathbf{Z}^{q^2+1}/\mathcal{L}_2$ is not cyclic when $l \geq 1$. The relation generation stage only gives lattice vectors in $\mathcal{L}_2$, which is far from the $\mathcal{L}_1$ if $l \geq 1$. Thus, we need to add more relations to $\mathcal{L}_2$ in order to get close to $\mathcal{L}_1$.

Since $\lambda^{q^2-1} = 1$, the vector $(q^2-1, 0, \cdots, 0)$ is automatically in $\mathcal{L}_2$. Let $\mathcal{L}_2^*$ be the lattice in $\mathbf{Z}^{q^2+1}$ generated by $\mathcal{L}_2$ and the following q^2 vectors

$$(0, q^{2k}-1, 0, \cdots, 0), \cdots, (0, 0, \cdots, 0, q^{2k}-1),$$

corresponding to the relations $(x+\alpha_i)^{q^{2k}-1} = 1$ modulo $f(x)$ for $\alpha_i \in \mathbf{F}_{q^2}$. It is clear that

$$\mathcal{L}_2^* = \mathcal{L}_2 + (q^{2k}-1)\mathbf{Z}^{q^2+1}.$$

The next resultgives the group structure for the quotient $\mathbf{Z}^{q^2+1}/\mathcal{L}_2^*$.

Theorem 2 *For* $deg(h_1) \leq 2$*, there is a group isomorphism*

$$\mathbf{Z}^{q^2+1}/\mathcal{L}_2^* \cong \mathbf{Z}/(q^{2k}-1)\mathbf{Z} \oplus \bigoplus_{1 \leq i \leq l} \mathbf{Z}/(q^{2\gcd(k,k_i)}-1)\mathbf{Z}.$$

Proof: Recall that

$$\mathbf{Z}^{q^2+1}/\mathcal{L}_2 \cong A \stackrel{\text{def}}{=} \mathbf{Z}/(q^{2k}-1)\mathbf{Z} \oplus \bigoplus_{i=1}^{l} \mathbf{Z}/(q^{2k_i}-1)\mathbf{Z} \bigoplus (\text{a finite } p\text{-group}).$$

It is clear that

$$A/(q^{2k}-1)A \cong \mathbf{Z}/(q^{2k}-1)\mathbf{Z} \oplus \bigoplus_{1 \leq i \leq l} \mathbf{Z}/(q^{2\gcd(k_i,k)}-1)\mathbf{Z}.$$

The kernel of the surjective composed homomorphism

$$\mathbf{Z}^{q^2+1} \longrightarrow \mathbf{Z}^{q^2+1}/\mathcal{L}_2 \cong A \longrightarrow A/(q^{2k}-1)A$$

is precisely $\mathcal{L}_2 + (q^{2k}-1)\mathbf{Z}^{q^2+1} = \mathcal{L}_2^*$. The desired isomorphism follows.

□

If $\gcd(k_i, k) > 1$ for some i, then $\mathcal{L}_2^*$ is still far from $\mathcal{L}_1$. We would like $\mathcal{L}_2^*$ to be as close to $\mathcal{L}_1$ as possible in a smooth sense. For us, the more interesting case is the following

Corollary 2 *Let $deg(h_1) \leq 2$. If $\gcd(k_i, k) = 1$ for all $1 \leq i \leq l$, we have an isomorphism*

$$\mathbf{Z}^{q^2+1}/\mathcal{L}_2^* \cong \mathbf{Z}/(q^{2k}-1)\mathbf{Z} \oplus (\mathbf{Z}/(q^2-1)\mathbf{Z})^l.$$

This corollary shows that under the same assumption, the lattice $\mathcal{L}_2^*$ is a smooth approximation of $\mathcal{L}_1$ in the sense that the quotient $\mathcal{L}_1/\mathcal{L}_2^*$ is a direct sum of small order cyclic groups.

The algorithm to compute the discrete logarithms in the factor base essentially samples vectors from the lattice $\mathcal{L}_2$. Let $\mathbf{r}_1, \mathbf{r}_2, \ldots,$ be the vectors in $\mathcal{L}_2$ obtained by the relation-finding algorithm, i.e., from the relations in (4). Let $\hat{\mathcal{L}}_2$ be the lattice generated by those vectors. Let $\hat{\mathcal{L}}_1$ be the lattice generated by $\hat{\mathcal{L}}_2$ and the following q^2+1 vectors:

$$(q^2-1, 0, \cdots, 0), (0, q^{2k}-1, 0, \cdots, 0), \cdots, (0, 0, \cdots, 0, q^{2k}-1).$$

Computing the Hermite (or Smith) Normal Form of $\hat{\mathcal{L}}_1$ is equivalent to solving the linear system $\hat{\mathcal{L}}_2$ in the ring $\mathbf{Z}/(q^{2k}-1)\mathbf{Z}$. It is in general difficult to find bases for the two lattices $\mathcal{L}_1$ and $\mathcal{L}_2$ directly. One can think that $\hat{\mathcal{L}}_1$ and $\hat{\mathcal{L}}_2$ are the approximations of $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively. These approximations can be computed by the polynomial time algorithm. Obviously,

$$\begin{array}{l} \hat{\mathcal{L}}_2 \subseteq \mathcal{L}_2 \subseteq \\ \qquad\quad \subseteq \hat{\mathcal{L}}_1 \subseteq \end{array} \mathcal{L}_2^* \subseteq \mathcal{L}_1.$$

These inclusions induce surjective group homomorphisms

$$\begin{array}{l} \mathbf{Z}^{q^2+1}/\hat{\mathcal{L}}_2 \to \mathbf{Z}^{q^2+1}/\mathcal{L}_2 \to \\ \qquad\quad \to \mathbf{Z}^{q^2+1}/\hat{\mathcal{L}}_1 \to \end{array} \mathbf{Z}^{q^2+1}/\mathcal{L}_2^* \to \mathbf{Z}^{q^2+1}/\mathcal{L}_1.$$

If $\mathbf{Z}^{q^2+1}/\hat{\mathcal{L}}_2$ is cyclic, then its quotient $\mathbf{Z}^{q^2+1}/\mathcal{L}_2$ will be cyclic. This is false if $l \geq 1$ as we have seen before. Similarly, $\mathbf{Z}^{q^2+1}/\hat{\mathcal{L}}_1$ is not cyclic as its quotient $\mathbf{Z}^{q^2+1}/\mathcal{L}_2^*$ is not cyclic if $l \geq 1$. Hence the conjecture in [9] also needs modification. It seems reasonable to hope that $\hat{\mathcal{L}}_1$ is a good approximation to $\mathcal{L}_2^*$ in the sense that the quotient $\mathcal{L}_2^*/\hat{\mathcal{L}}_1$ is a direct sum of small order cyclic groups. In the interesting case when $\gcd(k, k_i) = 1$ for all $1 \leq i \leq l$, our numerical data suggest the following highly plausible

Heuristics 1 *Assume that $x^q h_1(x) - h_0(x)$ does not have linear factors, and $\gcd(k, k_i) = 1$ for all $1 \leq i \leq l$. Then in the Smith Normal Form of $\hat{\mathcal{L}}_1$, the diagonal elements are*

$$1, 1, \cdots, 1, s_1, \cdots, s_t, q^{2k}-1,$$

where for $1 \leq i \leq t$, $s_i > 1$ and $s_i | q^2-1$.

Assuming the heuristics, $\mathbf{Z}^{q^2+1}/\hat{\mathcal{L}}_1$ is not much bigger than $\mathbf{Z}^{q^2+1}/\mathcal{L}_1$, namely,

$$\mathbf{Z}^{q^2+1}/\hat{\mathcal{L}}_1 \cong \mathbf{Z}/s_1\mathbf{Z} \oplus \mathbf{Z}/s_2\mathbf{Z} \oplus \cdots \oplus \mathbf{Z}/s_t\mathbf{Z} \oplus \mathbf{Z}/(q^{2k}-1)\mathbf{Z}.$$

We can find a generator for each component, as a product of linear polynomials from the computation of the Smith Normal Form. Suppose that for $1 \le i \le t$, the generator for the i-th component is

$$\lambda^{e_{i0}} \prod_{1 \le j \le q^2} (x+\alpha_j)^{e_{ij}}.$$

Since $s_i | q^2-1$, the above generator belongs to $\mathbf{F}_{q^2}$ in $\mathbf{F}_{q^2}[x]/(f(x))$. Assuming that it is $\lambda^{e'_{i0}}$, we have

$$\lambda^{e_{i0}-e'_{i0}} \prod_{1 \le j \le q^2} (x+\alpha_j)^{e_{ij}} = 1 \pmod{f(x)}.$$

There are t such relations. Adding them to $\hat{\mathcal{L}}_1$, we will finally arrive at the lattice $\mathcal{L}_1$. It allows us to find a generator for $(\mathbf{F}_{q^2}[x]/(f(x)))^*$, and to solve the discrete logarithms for the factor base, w.r.t. this generator.

4 The trap to the QPA-descent

Now we review the QPA-descent. Suppose that we need to compute the discrete logarithm of $W(\zeta) \in \mathbf{F}_{q^{2k}}[\zeta]$, where W is a polynomial over $\mathbf{F}_{q^2}$ of degree $w > 1$. The QPA-descent, firstly proposed in [3], is to represent $W(\zeta)$ as a product of elements of smaller degree, e.g. $\le w/2$, in the field $\mathbf{F}_{q^2}[x]/(f(x))$. To do this, one again starts with the identity:

$$\prod_{\alpha \in \mathbf{F}_q} (x-\alpha) = x^q - x.$$

Then apply the transformation

$$x \mapsto \frac{aW(x)+b}{cW(x)+d}$$

where the matrix $m = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathbf{F}_{q^2}^{2\times 2}$ is nonsingular. We have

$$\prod_{\alpha \in \mathbf{F}_q} \left(\frac{aW(x)+b}{cW(x)+d} - \alpha\right) = \left(\frac{aW(x)+b}{cW(x)+d}\right)^q - \frac{aW(x)+b}{cW(x)+d}.$$

Clearing the denominator:

$$\begin{aligned}
& (cW(x)+d)\prod_{\alpha\in\mathbf{F}_q}((aW(x)+b)-\alpha(cW(x)+d)) \\
= \;& (aW(x)+b)^q(cW(x)+d)-(aW(x)+b)(cW(x)+d)^q \\
= \;& (a^q\tilde{W}(x^q)+b^q)(cW(x)+d)-(aW(x)+b)(c^q\tilde{W}(x^q)+d^q),
\end{aligned}$$

where $\tilde{W}(x)$ is a polynomial obtained by raising the coefficients of $W(x)$ to the q-th power. Replacing x^q with $h_0(x)/h_1(x)$, we obtain

$$\begin{aligned}
& (cW(x)+d)\prod_{\alpha\in\mathbf{F}_q}((aW(x)+b)-\alpha(cW(x)+d)) \\
= \;& (a^q\tilde{W}(h_0(x)/h_1(x))+b^q)(cW(x)+d) \\
& -(aW(x)+b)(\tilde{W}(h_0(x)/h_1(x))+d^q h_1(x)) \\
& \quad (\mathrm{mod}\ x^q h_1(x)-h_0(x)).
\end{aligned}$$

It was shown in [3] that matrices in the same left coset of $PGL_2(\mathbf{F}_q)$ of $PGL_2(\mathbf{F}_{q^2})$ generate the same equations. The denominator of the right-hand size is a power of $h_1(x)$. Denote the numerator of the right-hand side polynomial by $N_{m,W}(x)$. If the polynomial $N_{m,W}(x)$ is $w/2$-smooth, namely, it can be factored completely into a product of irreducible factors over $\mathbf{F}_{q^2}$, all have degree $w/2$ or less, we obtain a relation of the form

$$\prod_{i=1}^{q^2}(W(x)+\alpha_i)^{e_i}=\lambda^{e_0}\prod_{g(x)\in S}g(x)^{e'_g}\pmod{x^q h_1(x)-h_0(x)}, \tag{5}$$

where $S\subseteq\mathbf{F}_{q^2}[x]$ is a set of monic polynomials of degrees less than $w/2$ and with cardinality at most $3w$. Denote the vector $(e_1,e_2,\ldots,e_{q^2})$ by $\mathbf{v}_m$. Note that it is a binary vector, and it is independent of $W(x)$. Collecting enough number of relations will allow us to represent $W(x)$ as a product of elements of smaller degrees. This process is the QPA-descent. A heuristic, made in [3], is that repeating the process, one can represent any element in $\mathbf{F}_{q^2}[x]/(f(x))$ as a product of linear factors. Combining it with the fact that the discrete logarithm of the linear factors are known, one solves the discrete logarithm for any element.

However the descent will not work if $W(x)$ is a factor of $x^q h_1(x)-h_0(x)$. Recall that $\alpha_1=0$.

Theorem 3 *If $W(x)|x^q h_1(x)-h_0(x)$, e_1 will always be 0 in (5).*

In other words, if $W(x)$ is a factor of $x^q h_1(x)-h_0(x)$, then it will never appear in the left-hand side of (5) as a factor. So the descent for $W(\zeta)$ is not possible.

Proof: The polynomial $W(x)$ is a zero divisor in the ring $\mathbf{F}_{q^2}[x]/(x^q h_1(x) - h_0(x))$. Hence if $W(x)$ appears in the left-hand side of (5) as a factor, it will also appear in the right-hand side. This contradicts the requirement that the factors in the right-hand side have degrees smaller than the degree of $W(x)$. □

Note that the trap factor $W(\zeta)$ can appear in the descent paths of other elements, which essentially blocks the descents. It is especially troublesome if $x^q h_1(x) - h_0(x)$ has many small degree factors.

5 The trap-avoiding descent

Now we have discovered traps for the original QPA-descent. How can we work around them? From the above discussion, we assume that we work in a non-Kummer extension, and the polynomial $x^q h_1(x) - h_0(x)$ with the factorization as (1) satisfies

- $\deg(h_0) \leq 2, \deg(h_1) \leq 1$;
- $k_i > 1$ for all $1 \leq i \leq l$; In other words, it is free of linear factors;
- $\gcd(k, k_i) = 1$ for all $1 \leq i \leq l$.

In the most interesting case when k is a prime, our numerical data show that the above requirements can be easily satisfied.

Heuristics 2 *Let q be a prime power and $k < q$ be a prime. Then there exist polynomials h_0 and h_1 satisfying the above requirements.*

Assume that the discrete logarithms of all linear polynomials have been computed. Suppose that we need to compute the discrete logarithm of $W(\zeta)$, where $W(x)$ is an irreducible polynomial of degree less than k, and it is relatively prime to $f(x)$. If $W(x)|x^q h_1(x) - h_0(x)$, we will search for an integer i such that $W(x)^i \pmod{f(x)}$ is relatively prime to $x^q h_1(x) - h_0(x)$. Such i can be found easily by a random process.

Now we can assume that $\gcd(W(x), x^q h_1(x) - h_0(x)) = 1$. If there are not many traps, we will use a trap-avoiding strategy for the descent. The basic idea is simple. Whenever we find a relation (5), we will not use it unless the right-hand side is relatively prime to $x^q h_1(x) - h_0(x)$.

Definition 1 *Define the trap-avoiding descent lattice $\mathcal{L}(W)$ associated with $W(x)$ to be generated by*

$$\{\mathbf{v}_m | N_{m,W} \text{ is } w/2 - \text{smooth, and } \gcd(N_{m,W}, x^q h_1(x) - h_0(x)) = 1\}.$$

Note that we use less relations than [3] does, since we have to avoid traps. If the vector $(1, 0, \ldots, 0)$ is in the trap-avoiding descent lattice of $W(x)$, then $W(x)$ can be written as a product of low degree polynomials in $\mathbf{F}_{q^2}[x]/(f(x))$ that are not traps. We believe that the following heuristics is very likely to be true.

Heuristics 3 *The trap-avoiding descent lattice for $W(x)$ contains the vector $(1, 0, \ldots, 0)$ if $\gcd(W(x), x^q h_1(x) - h_0(x)) = 1$.*

To provide a theoretical evidence, we will show that $(1, 0, \ldots, 0)$ is in its super lattice that is generated by $\mathbf{v}_m$ for all $m \in \mathcal{P}_q$, regardless whether $N_{m,W}(x)$ is $w/2$-smooth or not. This is a slight improvement over [3], where it is proved that $(q^3 - q, 0, \ldots, 0)$ is in the super lattice. To proceed, we first make some definitions following [3]. There are two matrices in consideration. The matrix $\mathcal{H}$ is composed by the binary row vectors $\mathbf{v}_m$ for all $m = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathcal{P}_q$. It is a matrix of $q^3 + q$ rows and q^2 columns. If we view m^{-1} as a map from $\mathbf{P}^1(\mathbf{F}_q)$ to $\mathbf{P}^1(\mathbf{F}_{q^2})$ given by

$$(\beta_1 : \beta_2) \to (-d\beta_1 + b\beta_2 : c\beta_1 - a\beta_2),$$

then the i-th component of v_m is 1 iff there is a point $P \in \mathbf{P}^1(\mathbf{F}_q)$ such that $m^{-1}(P) = (\alpha_i : 1)$. We define a binary vector $\mathbf{v}_m^+ = (e_1, \ldots, e_{q^2}, e_{q^2+1})$ for $m \in \mathcal{P}_q$, where $(e_1, \ldots, e_{q^2}) = \mathbf{v}_m$, and

$$e_{q^2+1} = \begin{cases} 1 & \text{if } (a : c) \in \mathbf{P}^1(\mathbf{F}_q) \\ 0 & \text{otherwise.} \end{cases}$$

One can verify that the last component of $\mathbf{v}_m^+$ corresponds to whether there is a point $P \in \mathbf{P}^1(\mathbf{F}_q)$ such that $m^{-1}(P) = (1 : 0) = \infty$. The matrix $\mathcal{H}^+$ is composed by the vectors $\mathbf{v}_m^+, m \in \mathcal{P}_q$. $\mathcal{H}^+$ is a matrix of $q^3 + q$ rows and $q^2 + 1$ columns. All the row vectors have exactly $q + 1$ many coordinates which are 1's.

Denote the lattices generated by the row vectors of $\mathcal{H}$ and $\mathcal{H}^+$ by $\mathcal{L}(\mathcal{H})$ and $\mathcal{L}(\mathcal{H}^+)$ respectively. In [3], the authors showed that $\mathbf{v}_1 = (q^2 + q, \ldots, q^2 + q) \in \mathcal{L}(\mathcal{H}^+)$ and $\mathbf{v}_2 = (q^2 + q, q + 1, \ldots, q + 1) \in \mathcal{L}(\mathcal{H}^+)$.

Theorem 4 *The vector $(1, 0, \ldots, 0)$ is in the lattice $\mathcal{L}(\mathcal{H})$.*

Proof: Fix a γ such that $\mathbf{F}_{q^2} = \mathbf{F}_q[\gamma]$. Firstly, observe that $\mathbf{v}_3 = (1, \ldots, 1, q) \in \mathcal{L}(\mathcal{H}^+)$. This follows from $\mathbf{v}_3 = \sum_{\beta \in \mathbf{F}_q} \mathbf{v}_{m_\beta} \in \mathcal{L}(\mathcal{H}^+)$, where $m_\beta = \begin{pmatrix} 1 & \beta\gamma \\ 0 & 1 \end{pmatrix} \in \mathcal{P}_q$. There are $q + 1$ row vectors in $\mathcal{H}^+$ such that both the first and the last coordinates are 1. Since the projective linear map on a projective line is sharply

3-transitive, a third coordinate with value 1 will uniquely determine the coset in $\mathcal{P}_q$. Thus the sum of these $q+1$ vectors is $\mathbf{v}_4 = (q+1, 1, \ldots, 1, q+1) \in \mathcal{L}(\mathcal{H}^+)$.

From the above observations, we have

$$\mathbf{v}_5 = \mathbf{v}_2 - (q+1)\mathbf{v}_3 = (q^2 - 1, 0, \ldots, 0, 1 - q^2) \in \mathcal{L}(\mathcal{H}^+),$$

$$\mathbf{v}_6 = \mathbf{v}_4 - \mathbf{v}_3 = (q, 0, \ldots, 0, 1) \in \mathcal{L}(\mathcal{H}^+).$$

We deduce

$$\mathbf{v}_7 = q\mathbf{v}_6 - \mathbf{v}_5 = (1, 0, \ldots, 0, q^2 + q - 1) \in \mathcal{L}(\mathcal{H}^+),$$

which implies $(1, 0, \ldots, 0) \in \mathcal{L}(\mathcal{H})$. □

6 Concluding Remarks and Open problems

In this paper, we study the validation of the heuristics made in the quasi-polynomial time algorithm solving the discrete logarithms in the small characteristic fields [3]. We find that the heuristics are problematic in the cases of non-Kummer extensions. We propose a few modifications to the algorithm, including some extra requirements for the polynomials h_0 and h_1, and a trap-avoiding descent strategy. The modified algorithm relies on three improved heuristics.

Proposition 1 *If Heuristics 1, 2 and 3 hold, then the discrete logarithm problem over $\mathbf{F}_{q^k}$ $(k < q)$ can be solved in time $q^{O(\log(k))}$.*

We believe that proving (or disproving) them are interesting open problems that help to understand the effectiveness of the new algorithm.

References

[1] L. M. Adleman. A subexponential algorithm for the discrete logarithm problem with applications to cryptography. In *Proc. 20th IEEE Symp. on Foundations of Comp. Science*, pages 55–60, 1979.

[2] Leonard M. Adleman. The function field sieve. In Leonard M. Adleman and Ming-Deh A. Huang, editors, *ANTS*, volume 877 of *Lecture Notes in Computer Science*, pages 108–121. Springer, 1994.

[3] Razvan Barbulescu, Pierrick Gaudry, Antoine Joux, and Emmanuel Thomé. A quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. Cryptology ePrint Archive, Report 2013/400, 2013.

[4] Don Coppersmith. Fast evaluation of logarithms in fields of characteristic two. *IEEE Transactions on Information Theory*, 30(4):587–594, 1984.

[5] W. Diffie and M.E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[6] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 33:469–472, 1985.

[7] Faruk Göloglu, Robert Granger, Gary McGuire, and Jens Zumbrägel. On the function field sieve and the impact of higher splitting probabilities. In Ran Canetti and Juan A. Garay, editors, *CRYPTO*, volume 8043 of *Lecture Notes in Computer Science*, pages 109–128. Springer, 2013.

[8] Daniel M. Gordon. Discrete logarithms in GF(p) using the number field sieve. *SIAM Journal on Discrete Mathematics*, 6(1):124–138, 1993.

[9] Ming-Deh Huang and Anand Kumar Narayanan. Finding primitive elements in finite fields of small characteristic. *CoRR*, abs/1304.1206, 2013.

[10] Antoine Joux. Faster index calculus for the medium prime case application to 1175-bit and 1425-bit finite fields. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT*, volume 7881 of *Lecture Notes in Computer Science*, pages 177–193. Springer, 2013.

[11] Antoine Joux. A new index calculus algorithm with complexity $L(1/4+o(1))$ in very small characteristic. Cryptology ePrint Archive, Report 2013/095, 2013.

[12] Antoine Joux and Reynald Lercier. The function field sieve in the medium prime case. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 254–270. Springer, 2006.

[13] Antoine Joux, Reynald Lercier, Nigel Smart, and Frederik Vercauteren. The number field sieve in the medium prime case. In *Advances in Cryptology - CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 326–344. Springer-Verlag, 2006.

[14] Ralph Merkle. *Secrecy, authentication, and public key systems*. PhD thesis, Stanford University, 1979.

[15] Daniel Panario, Xavier Gourdon, and Philippe Flajolet. An analytic approach to smooth polynominals over finite fields. In Joe Buhler, editor, *ANTS*, volume 1423 of *Lecture Notes in Computer Science*, pages 226–236. Springer, 1998.

[16] John Pollard. Monte carlo methods for index computations (mod p). *Mathematics of Computation*, 32(143):918–924, 1978.

[17] Daqing Wan. Generators and irreducible polynomials over finite fields. *Mathematics of Computation*, 66(219):1195–1212, 1997.