Index calculus for abelian varieties and the elliptic curve discrete logarithm problem

Pierrick Gaudry

Laboratoire LIX, École polytechnique, France
gaudry@lix.polytechnique.fr

March 4, 2004

Abstract. We propose an index calculus algorithm for the discrete logarithm problem on general abelian varieties. The main difference with the previous approaches is that we do not make use of any embedding into the Jacobian of a well-suited curve. We apply this algorithm to the Weil restriction of elliptic curves and hyperelliptic curves over small degree extension fields. In particular, our attack can solve all elliptic curve discrete logarithm problems defined over $\mathbb{F}_{q^3}$ in time $O(q^{10/7})$, with a reasonably small constant; and an elliptic problem over $\mathbb{F}_{q^4}$ or a genus 2 problem over $\mathbb{F}_{p^2}$ in time $O(q^{14/9})$ with a larger constant.

1 Introduction

The elliptic curve discrete logarithm problem is the key stone of the security of many cryptosystems [15, 19]. Except for a few families of weak curves [17, 26, 21, 24], the best known algorithms are generic algorithms, like Pollard's Rho algorithm [20] and its parallel variants [29]. Some attempts have been made to lift the problem, either to $\mathbb{Q}$, like in the Xedni algorithm [13, 25, 14], or to a local field [9]. None of them proved to be feasible. A more successful attack was based on the Weil restriction process [10, 6, 2, 12]: taking as input a discrete logarithm problem in an elliptic curve defined over an extension field, it is possible to transport it into the Jacobian of a curve of larger genus, but defined over a smaller base field than the initial field. Since there exist sub-exponential algorithms for discrete logarithms in Jacobians of high genus curves [1, 7, 5], in some cases this yields a faster attack than Pollard's Rho [18, 16].

Recently, Semaev posted two new attempts [22, 23] to solve the discrete logarithm problem on elliptic curves. Although they do not directly lead to complete algorithms, these papers are intriguing. In the present paper, we show that ideas taken from [23], mixed with a Weil restriction approach, combine into an algorithm that can solve the discrete logarithm on elliptic curves defined over small extension fields faster than Pollard's Rho. In particular, we shall demonstrate that a discrete log problem defined over a finite field of the form $\mathbb{F}_{p^3}$ can be solved in time $O(p^{10/7})$, which has to be compared with $O(p^{3/2})$ for Pollard's Rho. To obtain this complexity, we also make use of Thériault's large prime variant for low genus index calculus [27].

The paper is organized as follows: we start with an introductory example that explains an index calculus algorithm solving an elliptic curve discrete logarithm problem over $\mathbb{F}_{p^2}$ in time $O(p)$, which is as fast as Pollard's Rho. In Section 3, we give a general method to solve a discrete logarithm problem on an abelian variety. Then in Section 4, we use the Weil restriction method to apply our method to elliptic curves defined over extension fields. In that section, we shall see that Semaev's summation polynomials simplify the formulae. In

Section 5, we compare our method to the classical Weil descent attack, from a theoretical and practical point of view. Finally in Section 6, we apply our attack to hyperelliptic curves.

2 Introductory example: elliptic curves over $\mathbb{F}_{p^2}$

Let $p = 1019$. Then the polynomial $f(t) = t^2 + 1$ is irreducible over $\mathbb{F}_p$, and therefore $\mathbb{F}_{p^2}$ can be defined as $\mathbb{F}_p[t]/(t^2 + 1)$. Let E be the elliptic curve defined over $\mathbb{F}_{p^2}$ by $y^2 = x^3 + ax + b$, where

$$\begin{aligned} a &= a_0 + a_1 t = 214 + 364t, \\ b &= b_0 + b_1 t \ = 123 + 983t. \end{aligned}$$

It is easily checked that the group order of E is the prime $N = 1039037$. Let P be a random generator of E and Q a random point in E. For instance, take

$$P = (401 + 517t, 885 + 15t),$$

and

$$Q = (935 + 210t, 740 + 617t).$$

We define a factor base $\mathcal{F}$ for E to be the set of points of E that have an abscissa defined over $\mathbb{F}_p$. It has 1011 elements.

Let us form random linear combinations of P and Q and test if they can be written as the sum of two points in $\mathcal{F}$. For instance, let R be the point

$$R = 459328P + 313814Q = (415 + 211t, 183 + 288t).$$

Let $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ be two points in $\mathcal{F}$ such that $R = P_1 + P_2$. In [23], Semaev gives an explicit polynomial f_3 called a *summation polynomial* such that we must have $R = P_1 + P_2$ implies that $f_3(x_1, x_2, x_R) = 0$. Rewriting it in terms of $e_1 = x_1 + x_2$ and $e_2 = x_1 x_2$, we get

$$(e_1^2 - 4e_2)x_R^2 - 2(e_1 e_2 + ae_1 + 2b)x_R + a^2 + e_2^2 - 2ae_2 - 4be_1 = 0.$$

This equation relates quantities in $\mathbb{F}_{p^2}$ and the only unknowns are e_1 and e_2 that are required to be in $\mathbb{F}_p$. In order to convert this last requirement into an algebraic relation, we use the Weil restriction process, that is we explicitly have t enter the game. Hence, after reducing modulo $f(t)$, we obtain

$$(881e_1^2 + 597e_1e_2 + 31e_1 + 843e_2 + 669)\ t + (329e_1^2 + 189e_1e_2 + 971e_1 + e_2^2 + 294e_2 + 740) = 0.$$

For this equation to be verified, both coefficients in t must be zero. Therefore, we obtain two equations in two indeterminates over $\mathbb{F}_p$. Solving this system via resultants or Gröbner basis, we find the following possible value for (e_1, e_2):

$$(e_1, e_2) = (845, 1003).$$

And for this pair, we solve $(x - x_1)(x - x_2) = x^2 - e_1 x + e_2$. The solution we find is

$$x_1 = 92 \text{ and } x_2 = 753.$$

Then y_1 and y_2 are easily deduced, and we find

$$P_1 = (92, 779 + 754t) \text{ and } P_2 = (753, 628 + 692t).$$

After having produced 1012 such relations, we can solve a linear algebra problem to get a non-trivial combination of P and Q that is zero, and the discrete logarithm of Q in base P follows (we find $\log_P(Q) = 76982$).

In the main body of the paper, we will show that $\#\mathcal{F}$ is always of size about p, and that on average we obtain one relation for every two linear combinations of P and Q that we try. Therefore, the matrix of relations can be computed in time $O(p)$ up to logarithmic factors in p. Solving the linear system could be done using general sparse linear algebra tools like Lanczos's or Wiedemann's algorithms at a quadratic cost $O(p^2)$. In fact, since in each row we have only two non-zero entries, each step within the Gaussian elimination will maintain this sparseness. Hence it can be shown that the running time of the linear algebra step is in $O(p)$. We refer to [8] to a precise analysis of this ultra-sparse linear algebra, based on a graph interpretation.

Hence we have found an index-calculus algorithm that has the same complexity as Pollard-Rho up to logarithmic factors, namely $O(p)$ operations, but requires much more memory.

In the remainder of the paper, we shall formalize a generalization of this algorithm to abelian varieties, and we shall apply it to curves over $\mathbb{F}_{p^n}$ using Weil descent and analyze it for small values of $n \geq 3$.

3 An index calculus algorithm for abelian varieties

3.1 A convenient representation of abelian varieties

Let A be an abelian variety of dimension n, that is defined over a finite field $\mathbb{F}_q$ with q elements. We shall work with an explicit embedding of a dense Zariski-open subspace of A into an affine space of dimension $n+m$. In other words, an element $P \in A$ will be represented by $n + m$ coordinates

$$P = (x_1, \dots, x_n, y_1, \dots, y_m),$$

where x_i and y_i are in $\mathbb{F}_q$, and such a representation is possible for all the elements of A but a negligible proportion. Furthermore, we assume that for each choice of $x_1, \dots, x_n$ in $\overline{\mathbb{F}_q}$, there exist only finitely many m-uples $y_1, \dots, y_m$ in $\overline{\mathbb{F}_q}$ such that these $m + n$ coordinates yield a point of A.

In the case of dimension 1 where A is an elliptic curve, we can take for x_1 the classical abscissa coordinate and for y_i the ordinate. All the points except the point at infinity can be represented with these two coordinates. In the case where A is the Jacobian of a hyperelliptic curve, we can take for x_i the coefficients of the first polynomial in Mumford representation and for y_i the coefficients of the second polynomial. For general abelian varieties, no choice seems to be canonical, but usually the way A is constructed and its explicit group law already use such a coordinate system. Note also that a coordinate system with these properties is called a Noether normalization of the variety (see [4]).

The coordinates (x_i, y_i) of a point of A verify some equations that can be assumed to form a triangular set: the first equation is a polynomial in y_1 and the x_i, the second equation is a polynomial in y_1, y_2 and the x_i, and so on until the last equation which is a polynomial in all the coordinates. Such a triangular system makes explicit the fact that for each value of

x_i, there exist only finitely many m-uples for the y_i. This system has m equations and they locally define the variety A.

3.2 Definition of a factor base

Among all the points of A that can be represented in our coordinate system, we shall select some of them to form a factor base. For this, the field of definition of the coordinates is important.

We define the factor base $\mathcal{F}$ to be the following set

$$\mathcal{F} = \{P \in A \; ; \; P = (x_1, 0, 0, \ldots, 0, y_1, y_2, \ldots, y_m), \; x_1, y_1, \ldots, y_m \in \mathbb{F}_q\}.$$

Since the field $\mathbb{F}_q$ is finite, the factor base $\mathcal{F}$ is finite.

The factor base is then essentially a curve drawn on A, and we feel close to the classical Weil descent attack. However in our attack, we do not consider the Jacobian of the curve, so its genus is not directly involved.

3.3 Decomposing a point on the factor base

We want to address the following question:

Let P be a point on A. Are there points P_1, P_2, $\ldots$, P_n in $\mathcal{F}$ such that

$$P = P_1 + P_2 + \cdots + P_n,$$

and how to compute all the solutions?

The group law on A is defined by rational fractions in terms of the coordinates we use. Then there exist $n + m$ explicit rational fractions φ_1, $\ldots$, φ_{n+m} such that

$$P_1 + \cdots + P_n = (\varphi_1(P_1, \ldots, P_n), \ldots, \varphi_{n+m}(P_1, \ldots, P_n)).$$

Writing the equations corresponding to this $(m + n)$-uple being equal to P and also the equations describing the fact that all the points are indeed on A or in $\mathcal{F}$, we get a system with more equations than unknowns (i.e. the coordinates of P_1,..., P_n). The system is therefore (generically) of dimension 0, and has a finite number of solutions over $\overline{\mathbb{F}_q}$.

For a given P, finding all the solutions P_1, $\ldots$, P_n defined over $\mathbb{F}_q$, can be done by a Gröbner basis computation, followed by the factorization of a univariate polynomial. The degree of that polynomial is bounded by the degree of the ideal defined by all the equations that were in the system. The complexity of this computation is polynomial in the size of $\mathbb{F}_q$, and some parameters of the system that depend essentially on the degrees of the equations defining A (see [11]). In general, these parameters are exponential in n.

Remark 1. The rational fractions φ_1, $\ldots$, φ_{n+m} are usually valid only locally. For instance, evaluated at points with $P_1 = P_2$, one of them could degenerate into 0/0; just like for elliptic curves where the doubling formula is distinct from the adding formula. Averaged over all the points P in A, this non-universality of the rational fractions will make us lose a negligible quantity of decomposable points.

3.4 Index calculus computation

Let P be a point on A and let Q be another point that is a multiple of P. The goal is to compute the discrete logarithm of Q in base P.

Let a and b be random integers bounded by the order of P. We form the linear combination $R = aP + bQ$, and we try to decompose it on the factor base using the Gröbner basis approach of the previous paragraph. If we get a solution, we store it as a relation. It is possible that there are several solutions for a single R. In that case, we just get more relations for the same effort.

After having collected more relations than the cardinality of the factor base $\mathcal{F}$, a linear algebra elimination on the relations allows to generate a hopefully non-trivial linear combination between P and Q that is zero in A. The discrete logarithm is deduced readily.

3.5 Expected number of collected relations — Overall complexity

The key issue is how likely it is to find a relation.

Generically, the dimension of $\mathcal{F}$ is 1, therefore it is a curve or a union of curves. Since it is obtained as the intersection of an irreducible variety with hyperplanes, generically we obtain an irreducible curve. If this is not the case, we can try again after a random linear transformation of the coordinates. Hence, we can assume that $\mathcal{F}$ is an irreducible curve and therefore has a cardinality $\approx q$.

When decomposing a point P in A, we are precisely computing the preimage $f^{-1}(P)$ where f is the function defined from $\mathcal{F}^n/\mathfrak{S}_n$ to A by

$$f : (P_1, \dots, P_n) \mapsto P_1 + \cdots + P_n.$$

The expected number of elements in $f^{-1}(P)$ is then

$$\sum_{P \in A} \frac{f^{-1}(P)}{\#A} = \frac{1}{\#A} \#(\mathcal{F}^n/\mathfrak{S}_n).$$

Estimating that $\#A \approx q^n$, we obtain that the expected number of relations produced by each trial is $1/n!$.

This factorial is not a surprise, since in the classical Weil descent attack, a $g!$ factor occurs in the complexity.

This estimation is heuristic in that we did not ask that P generates the whole of A, so that the linear combinations $R = aP + bQ$ are not purely random in A. If we assume that P is a generator, and if we fix all the parameters of A (dimension, degrees of the equations), and let only the size of $\mathbb{F}_q$ grow to infinity, then the $1/n!$ estimate becomes more and more accurate.

The complexity of the algorithm can now be deduced, at least if one assumes that the parameters of A are fixed and q tends to infinity. Indeed, as mentioned above, the point decomposition can be done in polynomial time in $\log q$. The average number of obtained relations is in $1/n!$ which is a constant, so that we need $O(q)$ operations to collect the relations and then $O(q^2)$ steps to solve the linear algebra problem. This has to be compared with the complexity of the Pollard-Rho method which is in $O(q^{n/2})$.

Obviously, Thériault's improvements [27] to index-calculus on small genus curves also apply in our context, so that even for $n = 3$, we expect to have a complexity of $O(q^{10/7})$ instead of $O(q^{3/2})$.

We insist on the fact that the constant hidden in the $O()$ is big and grows very fast with n. We shall estimate it more precisely in the framework of the Weil descent of elliptic curves in the next section.

Remark 2. The analysis fails if all the points of $\mathcal{F}$ lie in a strict abelian subvariety B of A. Indeed, in that case all the combinations of points of $\mathcal{F}$ will lie in B, and $0 \in A$ will have a huge preimage by the function f defined above. The average number of relations of $1/n!$ is still valid, but the repartition is concentrated in one point, that is unlikely to be reached.

In that case, we can try again after a random linear transformation of the coordinates: the probability that $\mathcal{F}$ is still included in B is negligible.

4 Application to elliptic curves

Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$ with $q = p^n$, where p is a prime (or a prime power). Then, using the Weil descent approach, a discrete logarithm problem on E can be viewed as a discrete logarithm problem on an abelian variety of dimension n over $\mathbb{F}_p$.

We thus obtain the following result:

Theorem 1. *Let n be a fixed integer and let p be a prime (or a prime power) that we let grow to infinity. There exists an algorithm that can solve a discrete logarithm problem on any elliptic curve defined over a finite field with p^n elements in time $O(p^2)$ (up to logarithmic factors).*

With Thériault's improvements, we obtain an algorithm that runs in time $O(p^{2-\frac{4}{2n+1}})$ (up to logarithmic factors).

We shall show below that the constant hidden in the $O()$ grows very fast with n and only small degree extensions are vulnerable to this attack. Note that we used nowhere that the base field was a prime field, hence if the degree of the extension is composite, one can consider it as an extension of an intermediate subfield in order to keep n small.

In the remainder of this section, we give more details on the application to elliptic curves. In particular we show how Semaev's summation polynomials make the Gröbner basis a little more explicit, thus allowing to analyze the dependance in n of the complexity.

4.1 Semaev's summation polynomials

We recall here the definition and properties of the summation polynomials introduced by Semaev [23].

Definition 1. *Let E be an elliptic curve of equation $y^2 = x^3 + ax + b$. The* summation polynomials *f_n of E are defined by the following recurrence. The initial values for $n = 2$ and $n = 3$ are given by*

$$f_2(X_1, X_2) = X_1 - X_2$$

and

$$f_3(X_1, X_2, X_3) = (X_1-X_2)^2X_3^2-2((X_1+X_2)(X_1X_2+a)+2b)X_3+((X_1X_2-a)^2-4b(X_1+X_2)),$$

and for $n \geq 4$ and $1 \leq k \leq n-3$,

$$f_n(X_1, \dots, X_n) = \mathrm{Res}_X(f_{n-k}(X_1, \dots, X_{n-k-1}, X), f_{k+2}(X_{n-k}, \dots, X_n, X)).$$

Semaev proves that the apparent redundancy in the definition of f_n via different values of k is consistent. The raison d'être of these polynomials is the following result that relates f_n to the group law on E.

Theorem 2. *Let E/k be an elliptic curve, $n \geq 2$ an integer and f_n its n-th summation polynomial. Let $x_1, \dots, x_n$ be n elements of an algebraic closure $\overline{k}$ of k. Then $f_n(x_1, \dots, x_n) = 0$ if and only if there exists a n-tuple $(y_1, \dots, y_n)$ in $\overline{k}$, such that for all i, $P_i = (x_i, y_i)$ is a point of E and*

$$P_1 + \cdots + P_n = 0.$$

Furthermore, if $n \geq 3$, the polynomial f_n is symmetric of degree 2^{n-2} in each variable.

4.2 Explicit Weil restriction

Let E be an elliptic curve over $\mathbb{F}_{p^n}$, given by an equation $y^2 = x^3 + ax + b$.

We choose an explicit polynomial basis representation of $\mathbb{F}_{p^n}$ as an extension of $\mathbb{F}_p$: we take an irreducible monic polynomial $f(t)$ of degree n over $\mathbb{F}_p$, so that $\mathbb{F}_{p^n} = \mathbb{F}_p[t]/(f(t))$.

We define (an open subset of) the **Weil restriction** A of E as the set of $2n$-uples of elements $(x_0, \dots, x_{n-1}, y_0, \dots, y_{n-1})$ in $\mathbb{F}_p$ such that $x = x_0 + x_1 t + \cdots x_{n-1} t^{n-1}$ and $y = y_0 + y_1 t + \cdots y_{n-1} t^{n-1}$ are the coordinates of a point of E. The group law is inherited from the group law of E, thus turning A into an abelian variety of dimension n.

Then, a natural choice for the factor base is the set of points of A for which $x_1 = x_2 = \cdots = x_{n-1} = 0$, which correspond precisely to the points of E with abscissae defined over $\mathbb{F}_p$:

$$\mathcal{F} = \{P = (x, y) \in E;\ x \in \mathbb{F}_p\}.$$

It could be that this choice of $\mathcal{F}$ is not good, in the sense that $\mathcal{F}$ could be reducible. Then it is required to take another choice, for instance $x_0 = x_2 = \cdots = x_{n-1} = 0$. Hence $\mathcal{F}$ is no longer related to any Galois structure, so that we hope to avoid bad surprises. In the following, we assume that the first choice is appropriate.

The decomposition over the factor base as described above implies to write down a big system of equations that is solved using a Gröbner basis computation. This resolution is simplified a little using Semaev's summation polynomials.

Let R be a point of E that we want to write as a sum of n points $P_1, \dots, P_n$ whose abscissae are in $\mathbb{F}_p$. Writing $x_P = x_{0,P} + x_{1,P} t + \cdots x_{n-1,P} t^{n-1}$ for the abscissa of a point in E, we need to solve

$$f_{n+1}(x_{P_1}, x_{P_2}, \dots, x_{P_n}, x_R) = 0,$$

where x_R is known. We rewrite it as an equation between polynomials in t that we reduce modulo $f(t)$. Hence we obtain an equation of the form

$$\sum_{i=0}^{n-1} \varphi_i(x_{0,P_1}, \dots, x_{0,P_n})\ t^i = 0.$$

All these coefficients must be zero, so we get n equations in the n indeterminates $x_{0,P_1}, \dots, x_{0,P_n}$. Writing this system of equations is therefore immediate. Solving it is more complicated and we use Buchberger's algorithm for that task.

By construction, the system is symmetric. It pays off to symmetrize the equations before applying Buchberger's algorithm: we rewrite the polynomials φ_i in terms of the elementary symmetric polynomials $e_1, e_2, \dots, e_n$ of the variables $x_{0,P_1}, \dots, x_{0,P_n}$.

If we find solutions of the symmetric system defined over $\mathbb{F}_p$, then we look for rational roots of the corresponding polynomial to find the abscissae of the P_i (if there exists an $\mathbb{F}_p$-decomposition for R, then there exists a rational solution for the e_i, but the converse is false).

4.3 Degrees of the equations

To handle an elliptic curve discrete logarithm over $\mathbb{F}_{p^n}$, we need to use Semaev's summation polynomial f_{n+1}, which has degree 2^{n-1} in each variable. Once symmetrized, we obtain a system of n equations in the n indeterminates $e_1, \ldots, e_n$, each of them of total degree bounded by 2^{n-1}. Therefore the degree of the univariate polynomial in e_1 that we obtain in a lexicographic Gröbner basis is generically $2^{n(n-1)}$.

The cost of Buchberger's algorithm is polynomial in this degree, and so is the root finding algorithm that we have to apply to this polynomial.

The probability of finding one relation is $1/n!$, therefore the cost of finding one relation should also include a $n!$ factor. However, this factor is negligible compared to a polynomial in $2^{n(n-1)}$. Therefore the average cost for computing one relation is:

$$\mathrm{poly}\left(2^{n(n-1)}\right) \ \mathrm{poly}(\log p).$$

4.4 Example: $n = 3$

We ran a computer experiment to estimate the cost of the decomposition step. In practice, we used a few resultant computations instead of a full Gröbner basis computation. Then, the cost of the decomposition is about 100 ms on a Pentium IV, using Magma. This gives just a crude indication about what could be done for a real large scale computation: the resultants can certainly be optimized in several ways, taking into account the specific form of the polynomials; also it is in principle possible to compute the Gröbner basis generically once and for all, so that thereafter we just have to plug in the coordinates of the point R and proceed to the root-finding step.

Still we can not really hope to handle more than a hundred or a thousand decompositions per second on a single processor. For the sizes of p that are reachable with today's technology, this is clearly not enough to be faster than Pollard Rho, for which the basic operation is the elliptic curve addition, which can be carried out at a rate of 1 million per second. In that context, our complexity of $O(p^{1.43})$ will beat the complexity of Rho $O(p^{1.5})$ only for $p > 2^{100}$ (say), namely a size for which no experiment can be done.

5 Comparison with the classical Weil descent attack

We call "classical" Weil descent attack the algorithms that we can find in [10] where a curve $\mathcal{C}$ is drawn on the Weil restriction of the elliptic curve and then an index calculus is done in the Jacobian of $\mathcal{C}$. Therefore the genus g of $\mathcal{C}$ is the key value for evaluating the complexity. For the method to work, it is necessary to have $g \geq n$, but besides that condition, the smaller the genus is, the better the attack works.

5.1 Conceptual difference between the two attacks

The two attacks start in a similar way: one draws a curve $\mathcal{C}$ on A that is of small degree (in the classical Weil descent, there is a hope that taking a small degree yields a small genus). In our attack, the index calculus is then done directly between $\mathcal{C}^n/\mathfrak{S}_n$ and A, whereas in the classical Weil descent, the index calculus is done in the Jacobian of $\mathcal{C}$, that is between $\mathcal{C}^g/\mathfrak{S}_g$ and $\mathrm{Jac}(\mathcal{C})$, the discrete log having been transported into $\mathrm{Jac}(\mathcal{C})$ using the conorm map.

The following diagram illustrates the maps involved in the computation: on the left side is our attack, on the right side is the classical Weil descent attack.

$$\begin{tikzcd}
 & A \arrow[rrd, "\text{Map DL}"] & & \\
\mathcal{C}^n/\mathfrak{S}_n \arrow[ru, "(*)"] & \mathcal{C} \arrow[l] \arrow[u] \arrow[r] & \mathcal{C}^g/\mathfrak{S}_g \arrow[r, "(*)"'] & \mathrm{Jac}(\mathcal{C})
\end{tikzcd}$$

The arrows marked by $(*)$ are those where the index calculus takes place. In the classical Weil Descent, the fact that the abelian variety is an explicit Jacobian of a curve makes it easier than in our case where we have to use a Gröbner basis computation.

On the other hand, the probability of having a decomposable element is $1/n!$ versus $1/g!$.

5.2 Summary of Pros and Cons

Advantages of our method.

- Our method does not require any knowledge of the geometry of the curve $\mathcal{C}$. Nor is an explicit algorithm for working in the Jacobian needed.
- The factorial component in the complexity is always $n!$, as compared to $g!$, where $g \geq n$ can be exponential in n. Indeed, in [3], it is shown that this is the case if the curve in the Weil restriction is constructed in the same way as in [10].

Drawbacks of our method.

- Gröbner basis are not easy to deal with (but the ingredients of the classical Weil descent are not that easy either).
- If n is large, our attack does not allow to enlarge the factor basis: the limiting cost is not the $n!$ that comes from the choice for the smoothness bound, but the $2^{n(n-1)}$ that is inherent to the decomposition method. The only hope is that n is composite, so that we can use a smaller n on a larger subfield.

5.3 Comparison for $n = 3$.

In [10], there is an example of an elliptic curve over $\mathbb{F}_{p^3}$, for which a Weil descent attack was tried. The curve $\mathcal{C}$ that is found in the Weil restriction has genus 13, and there is no hint that it could be hyperelliptic. According to the work of Diem [3], the genus 13 is the best we can find with a GHS-attack for a generic curve. Therefore we can conclude that working in the

Jacobian of that curve is not a trivial task, and furthermore it is required to perform about $13! \approx 6\ 10^9$ operations in the Jacobian before finding a relation. Hence we take no risk in saying that finding a relation will be much more costly than with our method that computes a relation in about half a second, with a Magma implementation.

Furthermore, with a genus 13 curve, the complexity of the index-calculus will not beat the $O(p^{3/2})$ complexity of Pollard Rho, even using Thériault's trick that yields $O(p^{1.85})$.

On the other hand, in [3], Diem proved that there exist some elliptic curves over $\mathbb{F}_{p^3}$, such that the Weil restriction contains a curve of genus 3. For those particular curves, our attack is less efficient than the Diem's attack, since solving a Gröbner basis is more expensive than working in the Jacobian of a genus 3 curve.

6 Hyperelliptic curves

Let $\mathcal{C}$ be a hyperelliptic curve of genus g defined over $\mathbb{F}_{p^n}$, in the Jacobian of which we have a discrete logarithm problem to solve. The Weil restriction of the Jacobian of $\mathcal{C}$ is an abelian variety of dimension ng over $\mathbb{F}_p$, with an explicit group law in a system of coordinates inherited from Mumford's representation of divisors. Hence, by Section 3, we have an algorithm that runs in time $O(p^2)$ and even $O(p^{2-4/(2ng+1)})$ with Thériault's algorithm.

We now discuss how this general approach can be applied in practice and compared with previously known methods.

6.1 The case $n = 1$

In the case $n = 1$, we have no Weil restriction at all, and the abelian variety is the Jacobian itself. In that case, it is well known that there is an index-calculus algorithm based on the decomposition of divisors as sums of points [1, 7]. We explain now how this algorithm can be interpreted as a particular case of the algorithm we have presented in Section 3. We start by a slight change of coordinates: instead of using the Mumford representation for divisors, we multiply the first polynomial by a scalar, to make the constant term equal to 1. This is possible only if the support of the divisor does not include a point with a null abscissa. Hence, any divisor of the Jacobian except for a negligible proportion can be described with two polynomials

$$\langle u_g x^g + u_{g-1}x^{g-1} + \cdots + u_1 x + 1, v_{g-1}x^{g-1} + \cdots + v_1 x + v_0\rangle.$$

It is easy to check that we are in the conditions of Section 3, where the u_i coordinates play the role of the x_i and the v_i are for the y_i. We then define the factor base $\mathcal{F}$ to be the set of divisor for which $u_g = u_{g-1} = \cdots = u_2 = 0$. Hence $\mathcal{F}$ consists of the divisors whose support is just one point of the curve (and the point at infinity), that is precisely the factor basis in the classical index-calculus.

Now, for any divisor R in the Jacobian, one can try to write it as a sum of points $P_1 + \cdots + P_g$ of the factor base. In this particular case, the group law is such that the formal sum of the P_i divisors is extremely simple and does not involve any complicated rational fractions: the Gröbner basis phase is reduced to nothing, and we readily proceed to the factorization step.

Hence, the classical index calculus for Jacobian of hyperelliptic curves is a particular case of our algorithm for general abelian varieties, but with a choice of coordinates that is extremely favorable since the Gröbner basis computation disappears.

6.2 The case $n > 1$

For hyperelliptic curves defined over extension fields, it is also possible to make a choice of coordinates that makes the Gröbner basis computation easier. In a sense, we use the classical index-calculus mixed together with our algorithm.

We take the same variant of Mumford's representation as described in the previous section. The factor basis (after a Weil restriction), is the set of divisors for which $u_g = u_{g-1} = \cdots = u_2 = 0$ and u_1 is in $\mathbb{F}_p$. Then the decomposition can be done in two steps: first we try to write the given divisor R as a sum of n divisors $D_1 + D_2 + \cdots + D_n$, where the D_i are divisors for which all the u_i are in $\mathbb{F}_p$. Thereafter, each D_i is tested for smoothness by testing if its u-polynomial splits completely.

Hence, with that choice of coordinates, the Gröbner basis is made simpler: the formulae involve n times the group law instead of ng times. For instance, for genus 2 curves over $\mathbb{F}_{p^2}$, the decomposition step is clearly feasible in a reasonable amount of time (experiments are currently being done...) As a conclusion, those curves are much weaker than expected, since discrete logarithms can be computed in time $O(p^{14/9})$ with a reasonable constant.

We mention two related works in that direction. Arita has found a (non hyperelliptic) curve of genus 8 in the Weil restriction of any hyperelliptic curve of genus 2 over $\mathbb{F}_{p^2}$, thus obtaining a discrete logarithm running in time $O(p^{30/17})$. Thériault [28] has discovered some particular non-elliptic curves (including some hyperelliptic) for which the Weil restriction contains low genus curves, thus demonstrating weaknesses in the discrete logarithm problem on the corresponding Jacobians.

7 Conclusion

We have presented an attack of the elliptic curve discrete logarithm problem that combines ideas from Semaev's index calculus definition and from the Weil descent attack. We have shown that asymptotically, elliptic curves defined over small degree extension fields are weaker than those defined over prime fields or large prime degree extension fields. In particular we have proposed an algorithm to solve the discrete logarithm on elliptic curves defined over $\mathbb{F}_{p^3}$ in time $O(p^{1.43})$.

The framework we gave for this attack is quite general and it applies to all Jacobian of curves defined over small degree extension fields. For instance, we have an algorithm for computing discrete logarithms in Jacobians of genus 2 curves over $\mathbb{F}_{p^2}$ in time $O(p^{1.55})$.

Acknowledgements

Many thanks to Claus Diem, for his prompt answers to my questions and his helpful remarks. Thank you also to Andreas Enge who made a careful reading of an early version of this work.

References

1. L. M. Adleman, J. DeMarrais, and M.-D. Huang. A subexponential algorithm for discrete logarithms over the rational subgroup of the jacobians of large genus hyperelliptic curves over finite fields. In L. Adleman and M.-D. Huang, editors, *ANTS-I*, volume 877 of *Lecture Notes in Comput. Sci.*, pages 28–40. Springer–Verlag, 1994.

2. S. Arita. Weil descent of elliptic curves over finite fields of characteristic three. In T. Okamoto, editor, *Advances in Cryptolgy – ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Comput. Sci.*, pages 248–258. Springer-Verlag, 2000.
3. C. Diem. The GHS-attack in odd characteristic. *J. Ramanujan Math. Soc.*, 18:1–32, 2003.
4. D. Eisenbud. *Commutative algebra with a view toward algebraic geometry*, volume 150 of *Graduate Texts in Mathematics*. Springer-Verlag, 195.
5. A. Enge and P. Gaudry. A general framework for subexponential discrete logarithm algorithms. *Acta Arith.*, 102:83–103, 2002.
6. S. Galbraith and N. Smart. A cryptographic application of Weil descent. In *Cryptography and Coding, 7th IMA Conference*, volume 1746 of *Lecture Notes in Comput. Sci.*, pages 191–200. springer–Verlag, 1999. Full paper is HP-LABS Technical Report (Number HPL-1999-70).
7. P. Gaudry. An algorithm for solving the discrete log problem on hyperelliptic curves. In B. Preneel, editor, *Advances in Cryptology – EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Comput. Sci.*, pages 19–34. Springer–Verlag, 2000.
8. P. Gaudry. *Algorithmique des courbes hyperelliptiques et applications à la cryptologie*. PhD thesis, École polytechnique, 2000.
9. P. Gaudry. Some remarks on the elliptic curve discrete logarithm. Unpublished manuscript, 2004.
10. P. Gaudry, F. Hess, and N. Smart. Constructive and destructive facets of Weil descent on elliptic curves. *J. of Cryptology*, 15:19–46, 2002.
11. M. Giusti. Complexity of standard bases in projective dimension zero. In J. Davenport, editor, *EUROCAL'87*, volume 378 of *Lecture Notes in Comput. Sci.*, pages 333–335. Springer-Verlag, 1987.
12. F. Hess. The ghs attack revisited. In E. Biham, editor, *Advances in Cryptology – EUROCRYPT 2003*, volume 2656 of *Lecture Notes in Comput. Sci.*, pages 374–387. Springer-Verlag, 2003.
13. M.-D Huang, K. Kueh, and K.-S. Tan. Lifting elliptic curves and solving the elliptic curve discrete logarithm problem. In L. Adleman and M.-D. Huang, editors, *ANTS*, volume 877 of *Lecture Notes in Comput. Sci.*, pages 377–384. Springer–Verlag, 2000.
14. M. Jacobson, N. Koblitz, J. Silverman, A. Stein, and E. Teske. Analysis of the Xedni calculus attack. *Des. Codes Cryptogr.*, 20:41–64, 2000.
15. N. Koblitz. Elliptic curve cryptosystems. *Math. Comp.*, 48(177):203–209, January 1987.
16. M. Maurer, A. Menezes, and E. Teske. Analysis of the GHS Weil descent attack on the ECDLP over characteristic two finite fields of composite degree. *LMS Journal of Computation and Mathematics*, 5:127–174, 2002.
17. A. Menezes, T. Okamoto, and S. A. Vanstone. Reducing elliptic curves logarithms to logarithms in a finite field. *IEEE Trans. Inform. Theory*, 39(5):1639–1646, September 1993.
18. A. Menezes and M. Qu. Analysis of the Weil descent attack of Gaudry, Hess and Smart. In D. Naccache, editor, *Topics in Cryptology – CT-RSA 2001*, volume 2020 of *LNCS*, pages 308–318. Springer-Verlag, 2001.
19. V. Miller. Use of elliptic curves in cryptography. In A. M. Odlyzko, editor, *Advances in Cryptology – CRYPTO '86*, volume 263 of *Lecture Notes in Comput. Sci.*, pages 417–426. Springer-Verlag, 1987.
20. J. M. Pollard. Monte Carlo methods for index computation mod p. *Math. Comp.*, 32(143):918–924, July 1978.
21. T. Satoh and K. Araki. Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves. *Comment. Math. Helv.*, 47(1):81–92, 1998.
22. I. Semaev. A reduction of the space for the parallelized Pollard lambda search on elliptic curves over prime finite fields and on anomalous binary elliptic curves. Preprint, 2003.
23. I. Semaev. Summation polynomials and the discrete logarithm problem on elliptic curves. Preprint, 2004.
24. I. A. Semaev. Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curves in characteristic p. *Math. Comp.*, 67(221):353–356, January 1998.
25. J. Silverman. The Xedni calculus and the elliptic curve discrete logarithm problem. *Des. Codes Cryptogr.*, 20:5–40, 2000.
26. N. Smart. The discrete logarithm problem on elliptic curves of trace one. *J. of Cryptology*, 12(3):193–196, 1999.
27. N. Thériault. Index calculus attack for hyperelliptic curves of small genus. In C. Laih, editor, *Advances in Cryptology – ASIACRYPT 2003*, volume 2894 of *Lecture Notes in Comput. Sci.* Springer-Verlag, 2003.
28. N. Thériault. Weil descent attack for Kummer extensions. *J. Ramanujan Math. Soc.*, 18:281–312, 2003.
29. P. C. van Oorschot and M. J. Wiener. Parallel collision search with cryptanalytic applications. *J. of Cryptology*, 12:1–28, 1999.