Journal of Symbolic Computation 44 (2009) 1690–1702

Contents lists available at ScienceDirect

Journal of Symbolic Computation

journal homepage: www.elsevier.com/locate/jsc

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

Pierrick Gaudry [1]

LIX - École polytechnique, Route de Saclay, 91128 Palaiseau, France
LORIA, Campus Scientifique, BP 239, 54506 Vandoeuvre-Lès-Nancy, France

ARTICLE INFO

Article history:
Received 29 March 2007
Accepted 29 August 2008
Available online 30 November 2008

Keywords:
Discrete logarithm problem
Elliptic curve
Index calculus
Weil descent

ABSTRACT

We propose an index calculus algorithm for the discrete logarithm problem on general abelian varieties of small dimension. 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 an elliptic curve discrete logarithm problem defined over $\mathbb{F}_{q^3}$ in heuristic asymptotic running time $\tilde{O}(q^{4/3})$; and an elliptic problem over $\mathbb{F}_{q^4}$ or a genus 2 problem over $\mathbb{F}_{q^2}$ in heuristic asymptotic running time $\tilde{O}(q^{3/2})$.

1. Introduction

The elliptic curve discrete logarithm problem is the key stone of the security of many cryptosystems (Koblitz, 1987; Miller, 1987). Except for a few families of weak curves (Menezes et al., 1993; Smart, 1999; Satoh and Araki, 1998; Semaev, 1998), the best known algorithms are generic algorithms, like Pollard's Rho algorithm (Pollard, 1978) and its parallel variants (van Oorschot and Wiener, 1999). Some attempts have been made to lift the problem to $\mathbb{Q}$, like in the Xedni algorithm (Huang et al., 2000; Silverman, 2000; Jacobson et al., 2000). But this proved not to be feasible. On the other hand, an approach based on the Weil restriction process (Gaudry et al., 2002; Galbraith and Smart, 1999; Arita, 2000; Hess, 2003) produced important results: 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

E-mail addresses: pierrick.gaudry@loria.fr, gaudry@lix.polytechnique.fr.
URL: http://www.loria.fr/~gaudry/.
[1] Tel.: +33 3 83 59 20 62; fax: +33 3 83 27 83 19.

0747-7171/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jsc.2008.08.005

sub-exponential algorithms for discrete logarithms in Jacobians of high genus curves (Adleman et al., 1994; Enge and Gaudry, 2002; Couveignes, 2001; Heß, 2004; Diem, 2006; Diem and Thomé, 2008; Enge and Gaudry, 2007), in some cases this yields a faster attack than Pollard's Rho (Menezes and Qu, 2001; Maurer et al., 2002).

In 2004, Semaev posted a new attempt (Semaev, 2004) to solve the discrete logarithm problem on elliptic curves. However this does not directly lead to a complete algorithm. In the present article, we show that ideas taken from Semaev (2004), mixed with a Weil restriction approach, combine into an algorithm that can solve the discrete logarithm on elliptic curves defined over small extension fields asymptotically faster than Pollard's Rho. In particular, we shall give evidence that a discrete log problem defined over a finite field of the form $\mathbb{F}_{q^3}$ can be solved in time $\tilde{O}(q^{4/3})$, which has to be compared with $\tilde{O}(q^{3/2})$ for Pollard's Rho. To obtain this complexity, we make use of Thériault's large prime variant for low genus index calculus (Thériault, 2003), and the improvement of it using two large primes (Gaudry et al., 2007).

Our main algorithm is designed in a slightly more general setting, in order to cover some other interesting cases, like Jacobians of hyperelliptic curves. In fact we describe an index calculus algorithm that can in principle work in any abelian variety. However, we rely on a bound on the number and the degree of the equations describing the input abelian variety; therefore the analysis is valid only for varieties inside some particular families of abelian varieties. Fortunately, this covers all the cases that are considered for cryptography (Jacobian of curves, or Weil restrictions of them).

The paper is organized as follows. In Section 2, we give a general method to solve a discrete logarithm problem on an abelian variety, and make precise what we mean by giving an abelian variety as input to our algorithm. Then in Section 3, we use the Weil restriction method to apply our algorithm to elliptic curves defined over extension fields. In that section, we shall see that Semaev's summation polynomials simplify the formulae. In Section 4, we compare our method to the classical Weil descent attack, from a theoretical and practical point of view. Finally in Section 5, we apply our attack to hyperelliptic curves.

2. An index calculus algorithm for abelian varieties

A sketch of the algorithm is as follows:

(1) Take as input an abelian variety A and two points on it in a convenient representation;
(2) Randomize coordinates;
(3) Define a factor base;
(4) Compute relations;
(5) Combine relations with linear algebra.

We shall give details on each step and finally give a complexity estimate.

2.1. 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.

In order to talk about algorithms related to A, we need to tell precisely how A is given. A first choice is to stay as close as possible to the mathematical definition, and to ask for a description of A as a smooth projective variety and for an atlas defining the group law on a finite open covering of $A \times A$. This is the approach taken for instance in Pila (1990). However, quite often such a representation of A is not the best suited for computations. For instance, if A is the Jacobian of a curve of genus 2, the representation of A as a smooth projective variety requires a 15-dimensional embedding (Flynn, 1990). Therefore we prefer to take in input the abelian variety A in a form which is readily convenient for computations and that we describe in this subsection. In many practical situations where we could be interested in solving a discrete logarithm problem in A, the representation of A arises naturally in such a convenient form: this is the case for Jacobian of curves where elements are stored in Mumford representation and for Weil restrictions of these objects.

We now come to the precise definition of what we call a convenient representation of A. We refer the reader to Cox et al. (1997) for a reference on algebraic geometry with a computational perspective and to Eisenbud (1995) for more general results on commutative algebra.

In our work, we shall assume that A is given by an explicit embedding of a dense Zariski-open subspace of A into an affine space of dimension $n+m$. In other words, an element $P \in A$ defined over $\mathbb{F}_q$ will be represented by $n+m$ coordinates

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

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 $x_1, \ldots, x_n$ in $\overline{\mathbb{F}_q}$, there exist only finitely many m-tuples $y_1, \ldots, y_m$ in $\overline{\mathbb{F}_q}$ such that these $m+n$ coordinates yield a point of A. We assume also that the elements for which we are asked to solve the discrete logarithm problem are representable with the given coordinates. In our algorithm, we deal with many elements of A. If one is encountered that cannot be represented with our coordinates, the corresponding attempt to build a relation is discarded. This occurs very rarely and does not change the complexity.

The coordinates (x_i, y_i) of a point of A verify some equations that can be assumed to form a triangular set, that is to say: 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. With such a triangular system, the fact that for each value of x_i, there exist only finitely many m-tuples for the y_i becomes easily checked. This system has m equations and it locally defines the variety A.

In the following, we assume that we are given a discrete logarithm problem to solve in an abelian variety for which this convenient representation is known, together with maps for the group law in this coordinate system. We shall be interested in the complexity in q only, therefore in our estimates, the parameters n, m and the degrees of the equations describing A are supposed to be constant.

Here are some typical examples:

- In the case of dimension 1 where A is an elliptic curve, we can take for (x_1, y_1) the classical Weierstrass coordinates. 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 (see Menezes et al. (1997)) 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 that gives a convenient representation is nothing but a Noether normalization of the variety (see Eisenbud (1995)). The triangular set of equations can be obtained as a reduced Gröbner basis for the lexicographical order of the equations defining A.
- The case where A is the Weil restriction of an elliptic curve will be studied more thoroughly in Section 3.

2.2. Definition of a factor base

At this point, we have A and the input points given in a convenient representation with coordinates x_i, y_i. We start by applying a random linear change of variables on the x_i coordinates. This does not change the properties of the representation, and the new triangular set of defining equations is deduced from the original one by just applying the change of coordinates. This random linear transform will allow us to say that "in general event E does not happen", meaning that the probability that it occurs is low, with respect to this change of variables.

We select some of the points of A to define the **factor base** $\mathcal{F}$ by

$$\mathcal{F} = \{P \in A \cap H_2 \cap H_3 \cap \cdots \cap H_n \,;\; P \text{ defined over } \mathbb{F}_q\},$$

where H_i is the hyperplane of equation $x_i = 0$.

Then $\mathcal{F} = \{(x_1, 0, \ldots, 0, y_1, \ldots, y_m) \in A \ ; \ x_1, y_i \in \mathbb{F}_q\}$ is an algebraic variety (intersection of algebraic varieties) of dimension 1, since $y_1, \ldots, y_m$ are algebraic over x_1, which is free.

The next step in our algorithm will be to test that $\mathcal{F}$ is an absolutely irreducible curve. This can be done by testing the absolute irreducibility of the first equation in the triangular defining set, that involves only x_1 and y_1. If the curve is not absolutely irreducible, we start again with a new change of coordinates.

Since we are cutting A which is absolutely irreducible by hyperplanes, the theorem of Bertini could help us in proving that the probability of getting an absolutely irreducible curve is high. However, the base field we are considering is finite, so that the classical statement does not apply. Since there are other parts, later in the algorithm that are of heuristic nature, we will make no effort in adapting Bertini's theorem to our purpose and we content ourselves with the heuristic that for large enough q, the probability of $\mathcal{F}$ being absolutely irreducible is high.

The number of points in $\mathcal{F}$ can then be estimated by Weil's bound: if $\mathcal{F}$ is smooth, the number of $\mathbb{F}_q$-rational points is $q + O(\sqrt{q})$, where the constant depends on the genus of $\mathcal{F}$, which can be bounded by a formula that depends only on the degrees in $x_1, y_1, \ldots, y_m$ of the equations defining A. Asking that $\mathcal{F}$ is smooth could be too restrictive; however, for large enough q, on an heuristic basis, it is very unlikely to get a curve with a number of singularities that is not negligible compared to q. So, in the unlikely case where one gets a curve $\mathcal{F}$ with not enough points, one starts again with a new coordinate change.

In what follows, we shall also need the fact that the closure of $\mathcal{F}$ is not included in a strict abelian subvariety of A; this could occur when A is not simple, which is a special case that is usually excluded when discussing discrete logarithm computations. But even when A is not simple, if $\mathcal{F}$ is included in a strict abelian subvariety of A, this will be easily detected during the algorithm, since the event of a successful decomposition (see below) will occur with a probability that is much smaller than what the theory predicts; we then make a random affine transformation of the x_i coordinates, and we try again with the corresponding new $\mathcal{F}$ (with high probability, $\mathcal{F}$ will be suitable).

2.3. Computation of relations

Let P and Q be the two points of A for which the discrete logarithm has to be computed. A **relation** is a linear combination of P and Q that is written as a sum of elements of the factor base $\mathcal{F}$. We concentrate on the cases where the number of elements of $\mathcal{F}$ that are summed is n, the dimension of A. Hence a relation is of the form:

$$R = aP + bQ = P_1 + \cdots + P_n,$$

where P_i is in $\mathcal{F}$, for $i = 1, \ldots, n$.

To construct a relation, we start by taking a and b two integers at random modulo the group order and compute $R = aP + bQ$. Then we want to compute, if they exist, some corresponding points $P_1, \ldots, P_n$ in $\mathcal{F}$.

Let $\mathfrak{S}_n$ be the nth symmetric group. We introduce the map ψ from $\mathcal{F}^n/\mathfrak{S}_n$ to A defined by

$$\psi : (P_1, \ldots, P_n) \mapsto P_1 + \cdots + P_n.$$

Since $\mathcal{F}$ is not included in a proper abelian subvariety of A, the dimension of the image of ψ in A is n. Hence for a generic point R in A, the number of preimages by ψ over the algebraic closure of $\mathbb{F}_q$ is finite.

We now make this explicit. 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 $\varphi_1, \ldots, \varphi_{n+m}$ 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 $(n + m)$-tuple being equal to R 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 $P_1, \ldots, P_n$). The system is (generically) of dimension 0, since it has a finite number of solutions over $\overline{\mathbb{F}_q}$.

For a given R, 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.

Remark 1. The rational fractions $\varphi_1, \ldots, \varphi_{n+m}$ are valid only on a dense open subset of A^n. For instance, evaluated at points with $P_1 = P_2$, one of them could yield a division by 0; just like for elliptic curves where the classical doubling formula is distinct from the adding formula. Averaged over all the points in A, this non-universality of the rational fractions will make us lose a negligible quantity of decomposable points.

2.4. Combination of relations

Given P and Q, we assume that we have collected one more relation than the number of elements in $\mathcal{F}$. Let us add a j subscript to identify the data coming from the jth relation:

$$R_j = a_j P + b_j Q = \sum_{\mathfrak{p} \in \mathcal{F}} c_{\mathfrak{p},j}\, \mathfrak{p},$$

where $c_{\mathfrak{p},j}$ is a non-negative integer, and the sum of the $c_{\mathfrak{p},j}$ for a fixed j is equal to n.

The matrix $C = (c_{\mathfrak{p},j})$ has one more column than rows, and therefore there exists a non-zero vector v_j in its kernel. Then we can form the corresponding combination of relations:

$$\sum_j v_j R_j = \left(\sum_j v_j a_j\right) P + \left(\sum_j v_j b_j\right) Q = \sum_j \sum_{\mathfrak{p} \in \mathcal{F}} v_j c_{\mathfrak{p},j}\, \mathfrak{p}.$$

Exchanging the sums on the right-hand side, we see that we get 0. Therefore

$$\left(\sum_j v_j a_j\right) P + \left(\sum_j v_j b_j\right) Q = 0.$$

And the discrete logarithm can be deduced if $\sum_j v_j b_j \neq 0$ which happens with high probability.

Therefore combining relations and deducing the discrete logarithm reduce to a sparse linear algebra question. We mention that actually this linear algebra step must be performed not over $\mathbb{Z}$ but modulo the order of P in A.

Remark 2. If P and Q do not generate the whole abelian variety A, then several problems can occur in an index calculus type discrete logarithm computation. Some probability estimates can be wrong since R is no longer a random element of A, and some loops can run forever. Using classical randomization techniques as in Enge and Gaudry (2002), these problems can be overcome, as long as the group structure of A is explicitly known.

2.5. Complexity estimate

We are going to estimate the complexity of the algorithm only in terms of q tending to infinity. It means that we consider a family of abelian varieties of fixed dimension n, given in a convenient representation as defined above, with the parameters of this representation being also fixed: the integer m and the degrees of the equations defining the varieties are fixed (or are bounded by constants). Therefore, our analysis is directed towards special families of abelian varieties (in the same spirit as in Pila (2005)), like, for instance, Jacobians of hyperelliptic curves of fixed genus, or Weil restrictions of elliptic curves over extensions fields of a fixed degree.

In this setting, we assume that the input of the algorithm is already in the convenient representation, so that we count no cost for that. The initial randomization of coordinates involves a number of operations in $\mathbb{F}_q$ that depends only on the family of abelian varieties we are considering and is therefore bounded by a constant. The cost is then polynomial in $\log q$.

It is then required to test the absolute irreducibility of the curve that defines the factor base. This can be done in time polynomial in $\log q$ using for instance the algorithm in Gao (2003).

The construction of the factor base is as follows: for each value of x_1 in $\mathbb{F}_q$, we substitute it in the equations defining A, together with $x_2 = x_3 = \cdots = x_n = 0$. Due the triangular form of the set of equations, solving for $y_1, \ldots, y_m$ is a matter of univariate polynomial factorization over $\mathbb{F}_q$. The degrees and the number of equations to solve is fixed, so that for each choice of x_1, the computation of the points in the factor base with this first coordinate can be done in polynomial time in $\log q$. Therefore, building the factor base costs $\tilde{O}(q)$.

We now come to the question of the cost of computing one relation. Given a point $R = aP + bQ$, finding a corresponding decomposition as a sum of n points in the factor base resorts to doing a Gröbner basis computation, followed by the factorization of a univariate polynomial. The number of equations, the number of variables, and the degree of the equations are bounded, for a fixed family of abelian varieties. Buchberger's algorithm involves a number of field operations that can be bounded in terms of these data. Therefore the Gröbner basis computation takes a time which is polynomial in $\log q$ and exponential in the other parameters that are constants, and can be put in the $O()$. The factorization step also takes a time polynomial in $\log q$, so that finding the decomposition of R as a sum of points in the factor base can be done in polynomial time in $\log q$.

Here, and below, we mention the use of Buchberger's algorithm for computing Gröbner basis. This is enough for our complexity estimates. In practice other algorithms, like Faugère's F4 or F5 (1999; 2002) could advantageously be chosen instead.

The next key issue is how likely it is to find a relation. When decomposing a point R in A, we are precisely computing the preimages $\psi^{-1}(R)$ where ψ is the function defined above. The expected number of elements in $\psi^{-1}(R)$ is then

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

By Weil's bound, the cardinality of A is about q^n. Since $\#\mathcal{F}$ is about q, we obtain that the expected number of relations produced by each trial is in $1/n!$ up to an error term which tends to 0 as q tends to infinity.

We can now put all these elements together to get the complexity of the full algorithm. The cost of the initial computations is polynomial in $\log q$. The cost of building a matrix of relations is in $\tilde{O}(q)$. And finally the cost of linear algebra is in $\tilde{O}(q^2)$, using Lanczos or Wiedemann's algorithm that takes advantage of the sparseness of the matrix. This has to be compared with the complexity of the Pollard-Rho method which is in $\tilde{O}(q^{n/2})$.

In order to improve the complexity, one can try to rebalance the cost of building the matrix and the cost of linear algebra. For that, we use large primes, in the same spirit as in Thériault's algorithm (2003) and its improvements (Gaudry et al., 2007). The idea is as follows. Some of the elements of the factor base $\mathcal{F}$ are selected (arbitrarily) to be genuine elements of the factor base, and the others become "large primes". In the phase of search of relations, only the one that involve at most two large primes are kept for later use. The other ones are discarded. On a heuristic base, if there are $O(q^{1-\frac{1}{n}})$ genuine factor base elements and the rest are large primes, the probability of finding a valid relation is decreased by a factor of $O(q^{1-\frac{2}{n}})$. After as many as $O(q^{2-\frac{2}{n}})$ relations have been collected, one can show that the large prime parts of the relations can be eliminated to produce $O(q^{1-\frac{1}{n}})$ relations that involve only genuine elements of the factor base. Thereafter, the linear algebra step takes only $\tilde{O}(q^{2-\frac{2}{n}})$. Hence using large primes, we have been able to transfer some of the cost of the linear algebra step to the cost of the relation search. We refer to Gaudry et al. (2007) for a precise description of this trick. We emphasize that the complexity estimate is heuristic, since one has to assume that the combinatorics work as in an ideal case, although we have no control on the probabilities when combining relations to eliminate large primes. This heuristic nature is already present in Gaudry et al. (2007).

Finally, we obtain the following heuristic complexity:

Heuristic result 3. *Let us consider a family $(A_i)_{i \geq 1}$ of abelian varieties of dimension $n \geq 2$ given by explicit equations of the same form, where the cardinality of the field of definition $\mathbb{F}_{q_i}$ of A_i tends to infinity. Then there exists a probabilistic algorithm that can solve discrete logarithm problems in an abelian variety*

A over $\mathbb{F}_q$ in that family in heuristic time $\tilde{O}(q^{2-\frac{2}{n}})$. The constant in the $\tilde{O}()$ depends on n and on the family, but not on q.

3. Application to elliptic curves

Let E be an elliptic curve defined over a finite field $\mathbb{F}_{q^n}$, where q 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 $\mathbb{F}_q$. Since for fixed n, the form of the equations defining the Weil restriction of E are always the same, we are in the context of abelian varieties in a family.

We thus obtain the following result:

Heuristic result 4. *Let $n \geq 2$ be a fixed integer and let q be a prime or a prime power that we let grow to infinity. There exists a probabilistic algorithm that can solve a discrete logarithm problem on any elliptic curve defined over a finite field with q^n elements in heuristic time $\tilde{O}(q^{2-\frac{2}{n}})$, where the constant depends on n.*

We shall show below that the constant hidden in the $O()$ grows very fast with n and only elliptic curves defined over small degree extensions of finite fields are vulnerable to this attack. Note that since we allow the base field to be a non-prime field, 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 are a first step in the direction of a Gröbner basis, thus allowing to analyze the dependence in n of the complexity. For simplicity, we restrict to the case where the characteristic is larger than 3. Otherwise, the equations should be adapted accordingly.

3.1. Semaev's summation polynomials

We recall here the definition and properties of the summation polynomials introduced by Semaev (2004).

Definition 5. 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

$$\begin{aligned} f_3(X_1, X_2, X_3) &= (X_1 - X_2)^2 X_3^2 - 2((X_1 + X_2)(X_1X_2 + a) + 2b)X_3 \\ &\quad + ((X_1X_2 - a)^2 - 4b(X_1 + X_2)), \end{aligned}$$

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

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

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 6 (*Semaev*). *Let E be an elliptic curve defined over k, $n \geq 2$ an integer and f_n its nth summation polynomial. Let $x_1, \ldots, x_n$ be n elements of an algebraic closure $\bar{k}$ of k. Then $f_n(x_1, \ldots, x_n) = 0$ if and only if there exists a n-tuple $(y_1, \ldots, y_n)$ in $\bar{k}$, 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.

3.2. Explicit Weil restriction

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

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

We define (an open subset of) the **Weil restriction** A of E as the set of $2n$-tuples of elements $(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1})$ in $\mathbb{F}_q$ 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 corresponds precisely to the points of E with abscissae defined over $\mathbb{F}_q$:

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

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 pathological cases like $\mathcal{F}$ being an abelian subvariety of A if E was constructed by extension of scalars. 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 system of equations involves $n(n+1)$ indeterminates, namely the x_0 and the $(y_i)_{1 \le i \le n}$ coordinates of the n points in the decomposition. The use of Semaev's summation polynomials reduces this number of indeterminates to n, since the y_i coordinates are no longer involved. Hence the system of equations that will be obtained after the use of Semaev's polynomials can be seen as a set of generators for the elimination ideal of the original system that keeps only the variables x_i. Solving a system with less variables is certainly easier than solving an equivalent system with more variables and therefore we expect the use of Semaev's polynomials to be faster than a direct attempt to solve the system. We now give more details on this resolution.

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

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

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

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

where the φ_i are polynomials. All these coefficients must be zero, so we get n equations in the n indeterminates $x_{0,P_1}, \ldots, x_{0,P_n}$. 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, since this symmetrization reduces the degree of the ideal by a $n!$ factor. We rewrite the polynomials φ_i in terms of the elementary symmetric polynomials $e_1, e_2, \ldots, e_n$ of the variables $x_{0,P_1}, \ldots, x_{0,P_n}$.

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

3.3. Degrees of the equations

To handle an elliptic curve discrete logarithm over $\mathbb{F}_{q^n}$, we 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 reduced Gröbner basis is generically $2^{n(n-1)}$.

The cost of Buchberger's algorithm is at least 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 dependence in n in the complexity is at least a polynomial in $2^{n(n-1)}$.

3.4. A worked example for $n = 2$

We start with the smallest possible value $n = 2$. In that case, everything is simple enough to be written on paper, so we will give an explicit example to illustrate our algorithm.

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

$$a = a_0 + a_1 t = 214 + 364t,$$
$$b = b_0 + b_1 t = 123 + 983t.$$

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 $R = P_1 + P_2$. Rewriting the third summation polynomial in terms of $e_1 = x_1 + x_2$ and $e_2 = x_1 x_2$, we get

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

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 use the explicit definition of $\mathbb{F}_{p^2}$ as degree 2 extension of $\mathbb{F}_p$. Hence, after writing x_R, a and b as polynomials in t 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 from the equation of E, 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 $\log_P(Q) = 76982$).

3.5. Example: $n = 3$

We ran a computer experiment to estimate the cost of the decomposition step in the case $n = 3$. 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 an 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.

Still we cannot really hope to handle more than a hundred or a thousand decompositions per second on a single processor. For the sizes of q 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 $\tilde{O}(q^{1.33\cdots})$ will beat the complexity of Rho $\tilde{O}(q^{1.5})$ only for $q > 2^{65}$ (say), namely a size for which no experiment can be done, but which is commonly used in cryptography.

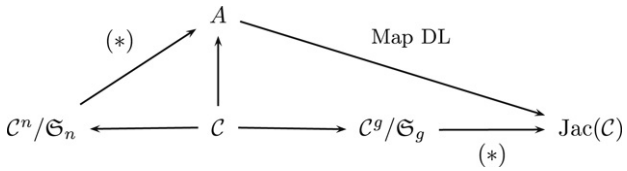
4. Comparison with the classical Weil descent attack

We call "classical" Weil descent attack the algorithms that we can find in Gaudry et al. (2002) 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. In the following, we assume that the reader is familiar with this algorithm.

4.1. 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 mapped 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.



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 $1/g!$.

4.2. Summary of pros and cons

4.2.1. 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 Diem (2003), it is shown that this is the case if the curve in the Weil restriction is constructed in the same way as in Gaudry et al. (2002).

4.2.2. 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.

4.3. Comparison for $n = 3$

In Gaudry et al. (2002), there is an example of an elliptic curve over $\mathbb{F}_{q^3}$, 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 (2003), for a generic elliptic curve over $\mathbb{F}_{q^3}$, the GHS-attack will produce curves of genus at least 13. 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 $13! \approx 8\ 10^9$ operations in the Jacobian before finding a relation. Hence 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 $\tilde{O}(q^{3/2})$ complexity of Pollard Rho, even using the improvements of Gaudry et al. (2007) that yield $\tilde{O}(q^{1.85})$.

On the other hand, in Diem (2003), Diem proved that there exist some elliptic curves over $\mathbb{F}_{q^3}$, such that the Weil restriction contains a curve of genus 3. For those particular curves, our attack is less efficient than Diem's attack, since solving a Gröbner basis is more expensive than working in the Jacobian of a genus 3 curve.

5. Hyperelliptic curves

Let $\mathcal{C}$ be a hyperelliptic curve of genus g defined over $\mathbb{F}_{q^n}$, 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 $\mathbb{F}_q$, with an explicit group law in a system of coordinates inherited from Mumford's representation of divisors. Hence, by Section 2, we have an algorithm that runs in heuristic time $\tilde{O}(q^{2-\frac{2}{ng}})$.

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

5.1. 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 (Adleman et al., 1994; Gaudry, 2000). We explain now how this algorithm can be interpreted as a particular case of the algorithm we have presented in Section 2. 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 2, 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.

5.2. 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 $\mathbb{F}_q$. 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 $\mathbb{F}_q$. 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 $\mathbb{F}_{q^2}$, the decomposition step is clearly feasible in a reasonable amount of time. As a conclusion, those curves are much weaker than expected, since discrete logarithms can be computed in time $\tilde{O}(q^{3/2})$ with a reasonable constant.

6. 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 $\mathbb{F}_{q^3}$ in heuristic time $\tilde{O}(q^{4/3})$.

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 $\mathbb{F}_{q^2}$ in heuristic time $\tilde{O}(q^{3/2})$.

Since this article has been made public as a preprint, some works have appeared that are based on it: Granger and Vercauteren (2005) have designed a variant that applies to algebraic tori; Nagao (2007) has improved the hyperelliptic case, making explicit the algebraic systems to solve, without any equivalent of Semaev's polynomials.

Acknowledgements

Many thanks to Claus Diem, for his prompt answers to my questions and his helpful remarks. Thanks are also to Andreas Enge who made a careful reading of an earlier version of this work and to Éric Schost for his help with the complexity of Gröbner basis computations.

References

Adleman, L.M., DeMarrais, J., Huang, M.-D., 1994. A subexponential algorithm for discrete logarithms over the rational subgroup of the jacobians of large genus hyperelliptic curves over finite fields. In: Adleman, L., Huang, M.-D. (Eds.), ANTS-I. In: Lecture Notes in Comput. Sci., vol. 877. Springer-Verlag, pp. 28–40.

Arita, S., 2000. Weil descent of elliptic curves over finite fields of characteristic three. In: Okamoto, T. (Ed.), Advances in Cryptolgy — ASIACRYPT 2000. In: Lecture Notes in Comput. Sci., vol. 1976. Springer-Verlag, pp. 248–258.

Couveignes, J.-M., 2001. Algebraic groups and discrete logarithm. In: Public-key Cryptography and Computational Number Theory. de Gruyter, pp. 17–27.

Cox, D., Little, J., O'Shea, D., 1997. Ideals, varieties, and algorithms. In: Undergraduate Texts in Mathematics, Springer-Verlag.

Diem, C., 2003. The GHS-attack in odd characteristic. J. Ramanujan Math. Soc. 18, 1–32.

Diem, C., 2006. An index calculus algorithm for plane curves of small degree. In: Pauli, S., Hess, F., Pohst, M. (Eds.), ANTS-VII. In: Lecture Notes in Comput. Sci., vol. 4076. Springer-Verlag, pp. 543–557.

Diem, C., Thomé, E., 2008. Index calculus in class groups of non-hyperelliptic curves of genus three. J. Cryptology 21 (4), 593–611.
Eisenbud, D., 1995. Commutative Algebra with a View Toward Algebraic Geometry. In: Graduate Texts in Mathematics, vol. 150. Springer-Verlag.
Enge, A., Gaudry, P., 2002. A general framework for subexponential discrete logarithm algorithms. Acta Arith. 102, 83–103.
Enge, A., Gaudry, P., 2007. An $L(1/3+\varepsilon)$ algorithm for the discrete logarithm problem for low degree curves. In: Naor, M. (Ed.), Advances in Cryptology — EUROCRYPT 2007. In: Lecture Notes in Comput. Sci., vol. 4515. Springer-Verlag, pp. 379–393.
Faugère, J.-C., 1999. A new efficient algorithm for computing Gröbner bases (F4). J. Pure Appl. Algebra 139, 61–88.
Faugère, J.-C., 2002. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). In: Mora, T. (Ed.), ISSAC '02. ACM Press, pp. 75–83.
Flynn, E.V., 1990. The Jacobian and formal group of a curve of genus 2 over an arbitrary ground field. Math. Proc. Cambridge Philos. Soc. 107, 425–441.
Galbraith, S., Smart, N., 1999. A cryptographic application of Weil descent. In: Cryptography and Coding, 7th IMA Conference. In: Lecture Notes in Comput. Sci., vol. 1746. Springer-Verlag, pp. 191–200. Full paper is HP-LABS Technical Report (Number HPL-1999-70).
Gao, S., 2003. Factoring multivariate polynomials via partial differential equations. Math. Comp. 72, 801–822.
Gaudry, P., 2000. An algorithm for solving the discrete log problem on hyperelliptic curves. In: Preneel, B. (Ed.), Advances in Cryptology — EUROCRYPT 2000. In: Lecture Notes in Comput. Sci., vol. 1807. Springer-Verlag, pp. 19–34.
Gaudry, P., Hess, F., Smart, N., 2002. Constructive and destructive facets of Weil descent on elliptic curves. J. Cryptology 15, 19–46.
Gaudry, P., Thomé, E., Thériault, N., Diem, C., 2007. A double large prime variation for small genus hyperelliptic index calculus. Math. Comp. 76, 475–492.
Granger, R., Vercauteren, F., 2005. On the discrete logarithm problem on algebraic tori. In: Shoup, V. (Ed.), Advances in Cryptology — CRYPTO 2005. In: Lecture Notes in Comput. Sci., vol. 3621. Springer-Verlag, pp. 66–85.
Hess, F., 2003. The GHS attack revisited. In: Biham, E. (Ed.), Advances in Cryptology — EUROCRYPT 2003. In: Lecture Notes in Comput. Sci., vol. 2656. Springer-Verlag, pp. 374–387.
Heß, F., Computing relations in divisor class groups of algebraic curves over finite, 2004, Preprint.
Huang, M.-D, Kueh, K., Tan, K.-S., 2000. Lifting elliptic curves and solving the elliptic curve discrete logarithm problem. In: Adleman, L., Huang, M.-D. (Eds.), ANTS. In: Lecture Notes in Comput. Sci., vol. 877. Springer-Verlag, pp. 377–384.
Jacobson, M., Koblitz, N., Silverman, J., Stein, A., Teske, E., 2000. Analysis of the Xedni calculus attack. Des. Codes Cryptogr. 20, 41–64.
Koblitz, N., 1987. Elliptic curve cryptosystems. Math. Comp. 48 (177), 203–209.
Maurer, M., Menezes, A., Teske, E., 2002. Analysis of the GHS Weil descent attack on the ECDLP over characteristic two finite fields of composite degree. LMS J. Comput. Math. 5, 127–174.
Menezes, A., Okamoto, T., Vanstone, S.A., 1993. Reducing elliptic curves logarithms to logarithms in a finite field. IEEE Trans. Inform. Theory 39 (5), 1639–1646.
Menezes, A., Qu, M., 2001. Analysis of the Weil descent attack of Gaudry, Hess and Smart. In: Naccache, D. (Ed.), Topics in Cryptology — CT-RSA 2001. In: LNCS, vol. 2020. Springer-Verlag, pp. 308–318.
Menezes, A., Wu, Y.-H., Zuccherato, R., 1997. An elementary introduction to hyper-elliptic curves. In: Koblitz, N. (Ed.), Algebraic Aspects of Cryptography. Springer-Verlag, pp. 155–178.
Miller, V., 1987. Use of elliptic curves in cryptography. In: Odlyzko, A.M. (Ed.), Advances in Cryptology — CRYPTO '86. In: Lecture Notes in Comput. Sci., vol. 263. Springer-Verlag, pp. 417–426.
Nagao, K., 2007. Decomposed attack for the jacobian of a hyperelliptic curve over an extension field. Cryptology ePrint Archive: Report 2007/112.
Pila, J., 1990. Frobenius maps of abelian varieties and finding roots of unity in finite fields. Math. Comp. 55 (192), 745–763.
Pila, J., 2005. Counting points on curves over families in polynomial time. Preprint available at: http://arxiv.org/abs/math.NT/0504570.
Pollard, J.M., 1978. Monte Carlo methods for index computation mod p. Math. Comp. 32 (143), 918–924.
Satoh, T., Araki, K., 1998. Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves. Comment. Math. Helv. 47 (1), 81–92.
Semaev, I., 2004. Summation polynomials and the discrete log algorithm problem on elliptic curves. Preprint, Available at: http://eprint.iacr.org/2004/031.
Semaev, I.A., 1998. Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curves in characteristic p. Math. Comp. 67 (221), 353–356.
Silverman, J., 2000. The Xedni calculus and the elliptic curve discrete logarithm problem. Des. Codes Cryptogr. 20, 5–40.
Smart, N., 1999. The discrete logarithm problem on elliptic curves of trace one. J. Cryptology 12 (3), 193–196.
Thériault, N., 2003. Index calculus attack for hyperelliptic curves of small genus. In: Laih, C. (Ed.), Advances in Cryptology — ASIACRYPT 2003. In: Lecture Notes in Comput. Sci., vol. 2894. Springer-Verlag, pp. 75–92.
van Oorschot, P.C., Wiener, M.J., 1999. Parallel collision search with cryptanalytic applications. J. of Cryptology 12, 1–28.