Speeding up the discrete log computation on curves with automorphisms

I. Duursma[1], P. Gaudry[2], and F. Morain[2] * **

[1] Université de Limoges, Laboratoire d'Arithmétique
Calcul formel et Optimisation,
123 avenue Albert Thomas,
F-87060 Limoges CEDEX, France
duursma@unilim.fr
[2] Laboratoire d'Informatique de l'École polytechnique (LIX)
F-91128 Palaiseau CEDEX, France
{gaudry, morain}@lix.polytechnique.fr
http://www.lix.polytechnique.fr/

Abstract. We show how to speed up the discrete log computations on curves having automorphisms of large order, thus generalizing the attacks on anomalous binary elliptic curves. This includes the first known attack on most of the hyperelliptic curves described in the literature.

1 Introduction

The use of elliptic curves in cryptography was first suggested by Miller [28] and Koblitz [19], following the work of Lenstra on integer factorization [23]. Many people improved on these ideas and the domain is flourishing. (See for instance the many books on that topic, e.g. [18, 27].) Koblitz [20] was the first to suggest using hyperelliptic curves for the same goal.

The security of many cryptosystems based on curves relies on the difficulty of the discrete log problem. In some special cases, it was shown that this problem was rather easy [26], [11], [44], [38], [40]. Apart from new lifting ideas [43, 17, 7, 6, 15] that remain to be tested, it seems that the discrete log on elliptic curves still resists. On ordinary curves, the only known attack is a parallelized version of Pollard's rho method [50]. The Certicom challenge[1] records the state of the art in the field.

Among the curves suggested for cryptographic use, the so-called Anomalous Binary Curves (ABC curves) have been shown to be somewhat less secure than ordinary curves, due to the existence of an automorphism of large order. Two types of attack have been suggested [52], [13]. The first one can be seen as an additive rho method, the second one as a multiplicative method.

---

* On leave from the French Department of Defense, Délégation Générale pour l'Armement.
** This work was supported by Action COURBES of INRIA (action coopérative de la direction scientifique de l'INRIA).
[1] See http://www.certicom.com/chal/.

The purpose of this article is to point out that these attacks can be generalized to the case where there exists an automorphim on the curve (resp. on the Jacobian of an hyperelliptic curve). In particular, we show how to obtain a speedup of $\sqrt{m}$ if there is an automorphism of order m.

The organization of the article is as follows. First of all, we recall Pollard's algorithm and its additive and multiplicative variants. Then, we describe the use of equivalence classes as originated in [52, 13]. In particular, we show how to tackle the problem of useless cycles in the Wiener-Zuccherato approach, by analyzing the number of useless cycles and being able to throw them out. We then show how to use automorphisms of large order on (hyper)elliptic curves, including generalized ABC curves [30, 45] and curves with CM by $\mathbb{Z}[\sqrt{-1}]$ or $\mathbb{Z}[(1+\sqrt{-3})/2]$, for which this appears to be the first published attack. This method applies also to hyperelliptic curves that were presented by Koblitz and others. We support our approach by numerical simulations.

2 Parallel collision search

Let G be a finite abelian group with law written additively $\oplus$ (and multiplication by k denoted by $[k]P$) and P an element of order n of G (w.l.o.g. we will assume that n is prime using the Pohlig-Hellman approach [33]). Let Q be an element of $< P >$, the cyclic group generated by P. The discrete log problem refers to the search for κ s.t. $Q = [\kappa]P$. For instance, G can be the group of points of an elliptic curve over a finite field, or the Jacobian of a hyperelliptic curve.

Generically, there exist three methods for computing discrete logs in time $O(\sqrt{n})$: Shanks's method [41], Pollard's rho and lambda methods [34], the last one being better when one has to find the discrete logarithm in a reduced range (that is not over $[0, n[$). The first method also uses $O(\sqrt{n})$ space, which is generally a problem. After recalling Pollard's rho, we show how a parallel collision search method can be described in the same spirit.

2.1 Pollard's rho method

It is well known that if we collect approximately k random values from a set of cardinality n, then we will find two equal values if $k = \sqrt{\pi n/2}$ on average. To solve the memory problem, Pollard [34] suggested to iterate a random function f on the group $< P >$. Starting from a point $R_0 \in < P >$, for instance $R_0 = [u_0]P \oplus [v_0]Q$ with u_0 and v_0 random integers (modulo n), one computes

$$R_{i+1} = f(R_i)$$

thereby obtaining two sequences (u_i) and (v_i) modulo n such that:

$$R_i = [u_i]P \oplus [v_i]Q.$$

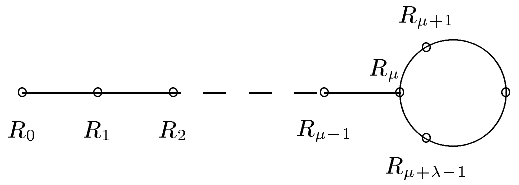

Fig. 1. The rho shape of (R_i).

In order to understand the properties of this random sequence, one is led to study the functional graph of f, that is the graph built using all starting points $R \in < P >$, as shown in Figure 1.

Such a sequence consists in a tail of length μ and a cycle of length λ, thus forming a rho of size $\rho = \mu + \lambda$. Asymptotic values for the critical parameters of this graph are given in [10]. For instance, a graph with n vertices should have $0.5 \log n$ (connected) components, the average tail (resp. cycle) length should be $\sqrt{\pi n/8}$, thus giving a rho length of about $\sqrt{\pi n/2}$.

These results imply that after about $\sqrt{\pi n/2}$ iterations, we will find a collision, that is two integers i and j for which $R_i = R_j$. This will give an identity

$$[u_i - u_j]P \oplus [v_i - v_j]Q = 0$$

from which we can deduce κ if $v_i \not\equiv v_j \bmod n$, which happens with probability $1 - 1/n$: Indeed, each element R of $< P >$ has n different representations as $[u]P \oplus [v]Q$ and the values v_i and v_j are supposed to be random integers in $[0, n-1]$.

2.2 Using distinguished points

An efficient implementation of parallel collision search requires storing a limited number of points that have a distinguished property (this idea was used for the first time in [35]). If θ is the proportion of distinguished points, then we should find a collision after computing about $\sqrt{\pi n/2}$ points, or $\theta\sqrt{\pi n/2}$ distinguished points. Now, parallelization is straightforward: have each processor contribute to the same list of distinguished points [50]. Each processor would have to find $\theta\sqrt{\pi n/2}/M$ points, thus yielding a speedup of M to the total running time.

To understand how a parallelized search works, consider figure 2, on which we have drawn two paths, the one corresponding to processor i and the second to processor j. Their paths collide at point C, which lies between the distinguished points D_1 and D_3 (resp. D_2 and D_3). The collision will be discovered as soon as we find that $R = D_3$.

Such a random walk has a very important feature: it is *deterministic*, which means that if $f(R) = C$, then $f(f(R)) = f(C)$, so that after hitting C from any direction, one follows a single path afterwards.

2.3 Choosing the function f

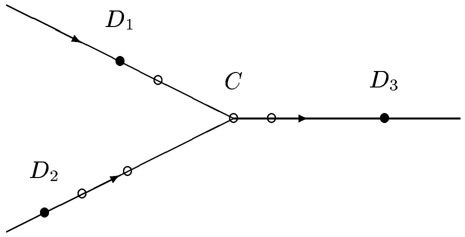

Fig. 2. The deterministic property.

An additive random walk. One way of choosing a good random function consists in precomputing r random points $(T^{(j)})_{0 \leq j < r}$ in $< P >$, with $T^{(j)} = [u^{(j)}]P \oplus [v^{(j)}]Q$ and to define the random walk as

$$f(R) = R \oplus T^{(j)}$$

where $j = \mathcal{H}(R)$ with $\mathcal{H}$ a hash function sending $< P >$ to $\{0, 1, \ldots, r-1\}$. This creates an *additive* random walk in $\mathbb{Z}/n\mathbb{Z}$ since each point R_i can be written as $R_i = [u_i]P \oplus [v_i]Q$ for which

$$u_{i+1} \equiv u_i + u^{(j)} \bmod n, \quad v_{i+1} \equiv v_i + v^{(j)} \bmod n.$$

Satler and Schnorr [39] have shown that the above approach is sufficiently random if $r \geq 8$. Teske has found experimentally [49] that a value of $r \geq 20$ is more convenient.

A multiplicative random walk. One can also use a function f built with fixed multipliers $(\mu_j)_{0 \leq j < r}$, μ_j defined modulo n. We define:

$$f(R) = [\mu_j]R$$

where $j = \mathcal{H}(R)$ as above. In this version, we would in fact compute:

$$f(R) = \left[\prod_{j=0}^{r-1} \mu_j^{f_j} \right] R_0.$$

A collision with a single sequence would not be interesting since this would lead to $(\prod_j \mu_j^{f_j}) \equiv 1 \bmod n$. A collision is interesting if it comes from two distinct sequences: two initial values $R_0 = [u_0]P \oplus [v_0]Q$ and $R_0' = [u_0']P \oplus [v_0']Q$ would lead to a more useful $[\prod_j \mu_j^{f_j}]([u_0]P \oplus [v_0]Q) = [u_0']P \oplus [v_0']Q$. For this to be random enough, we must have r large as above. To be efficient, the method requires that the multiples be efficiently computed.

Mixed walks. In practice, it would be fruitful to mix both strategies. This would lead to a walk that is difficult to analyze in theory but which is likely to be “more random”.

3 Random mappings on equivalence classes

Let $\sim$ be an equivalence relation on $< P >$ for which there are n/m equivalence classes. We note $\overline{R}$ for the equivalence class of R. If we can iterate Pollard's rho on the set of equivalence classes $< P > / \sim$, then we should find a collision in time $\sqrt{\pi n/(2m)}$ instead of $\sqrt{\pi n/2}$.

The easiest way of building an equivalence class is by using automorphisms of the group G. Wiener and Zuccherato [52] considered the case where G is the set of points of an elliptic curve E defined over a finite field $\mathbb{K}$. In that case, the equivalence relation would be $S \sim T$ if and only if $T = -S$. The first known nontrivial example of such a phenomenon came from ABC curves [52, 13] (see section 3.3 below).

More generally, if we know an automorphism α of order m operating on G, we can use the equivalence relation:

$$S \sim T \Longleftrightarrow \exists i, S = [\alpha^i]T$$

and hope to get a speedup of $\sqrt{m}$. We will give new examples in the following section.

From now on, we suppose our equivalence class is built on an automorphism α of order m. For the method to work efficiently, the equivalence class of a point should be easy to compute, that is much faster than an addition in G.

3.1 Well defined mappings

The first thing we can think of is to iterate f as usual, but perform the matches on the equivalence classes only. Starting from a random point R in $< P >$, we iterate $R = f(R)$; if $\overline{R} = \overline{R_j}$ for some j, try to find a match and stop, otherwise, store $\overline{R}$.

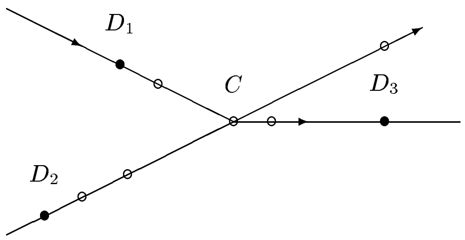

Fig. 3. Paths that cross each other.

This looks fine, as long as we do not want to parallelize the algorithm. In that case, we can come across the diagram of figure 3 which shows that the random walk would no longer be deterministic. For instance, imagine that we are again in the case of elliptic curves with an additive walk and that:

$$R_{i+1} = R_i + T^{(k)} = (x, y),$$

$$S_{j+1} = S_j + T^{(\ell)} = (x, -y).$$

In that case $\overline{S_{j+1}} = \overline{R_{i+1}}$, but in general $f(S_{j+1}) \neq f(R_{i+1})$.

Therefore, we must find a function f defined over $< P >$ and that is *well defined* over the equivalence classes. More precisely, we want that if $R' \in \overline{R}$, then $\overline{f(R')} = \overline{f(R)}$. The most obvious approach is to use $R_{i+1} = \overline{f(R_i)}$ and we iterate $R = \overline{f(R)}$ until a collision is found.

3.2 Useless cycles for additive walks

Let us explain why useless cycles appear. Suppose that G is the set $E(\mathbb{K})$ of points on an elliptic curve E over the finite field $\mathbb{K} = \mathbb{F}_{p^n}$. Suppose we use a function f as in section 2.3. Assume that the representative of a class is the point (x, y) with least y. Then we could encounter a situation where starting from R_i, we find

$$R_{i+1} = \overline{R_i \oplus T^{(j)}} = \ominus R_i \ominus T^{(j)}.$$

It could happen that

$$R_{i+2} = \overline{R_{i+1} \oplus T^{(j)}}$$

(the same j), thus yielding

$$R_{i+2} = \overline{\ominus R_i \ominus T^{(j)} \oplus T^{(j)}} = R_i.$$

This proves the existence of useless 2-cycles.

We fix some notations for studying these cycles. We consider a (primitive) t-cycle consisting of points $R_1, R_2, \ldots, R_t, R_{t+1} = R_1$. We denote by j_k the value $\mathcal{H}(R_k)$ of the hash function and we denote by $\varepsilon_k = \alpha^{e_k}$ the automorphism which gives the representative of the equivalence class of $\overline{R_k \oplus T^{(j_k)}}$. Thus we have $R_{k+1} = [\varepsilon_k](R_k \oplus T^{(j_k)})$. See figure 4 for a description of the cycle.

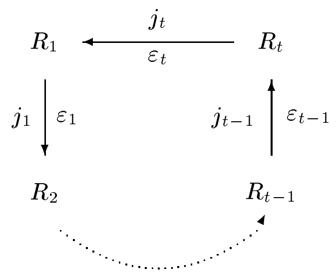

Fig. 4. A typical useless t-cycle.

We can now derive an expression of R_{t+1} in terms of R_1 and our parameters j_k and ε_k:

$$[1/\varepsilon_t]R_{t+1} = [\varepsilon_{t-1} \cdots \varepsilon_1]R_1 \oplus \tau$$

with $\tau = [\varepsilon_{t-1}\cdots\varepsilon_1]T^{(j_1)} \oplus [\varepsilon_{t-1}\cdots\varepsilon_2]T^{(j_2)} \oplus \cdots \oplus [\varepsilon_{t-1}]T^{(j_{t-1})} \oplus T^{(j_t)}$. We have a useless t-cycle if $\tau = 0$. Indeed, when $\varepsilon_1, \ldots, \varepsilon_{t-1}$ and $j_1, \ldots, j_t$ are chosen such that $\tau = 0$, then R_{t+1} is in the class of R_1, which does not impose any restriction on ε_t. The fact that the $T^{(j)}$ are randomly chosen implies that (very likely) there is no simple relation between them[2]. In that case, a necessary condition for having $\tau = 0$ is that for all $k \in \{1, 2, \ldots, t\}$ there exists a $l \neq k$ such that $j_k = j_l$.

Now, we can summarize the following results on the expected number of useless t-cycles.

Proposition 31 *Suppose there is an automorphism α of order m acting on a group G with n elements and that the $T^{(j)}$ are not related by relations involving powers of α. We can estimate the probability $\mathcal{P}(t)$ of useless t-cycles for small t when iterating an additive function with r branches (the expected number of t-cycles being $\mathcal{P}(t)(n/m/t)$).*

We have

$$\mathcal{P}(t) \leq \sum_{k=1}^{\min(r,t/2)} \frac{1}{m^k} \min\left(1, \frac{r!k^t}{(r-k)!r^t}\right).$$

Moreover, for small values of t we have the following bounds:

t	*pattern*	$\mathcal{P}(t)$	
2	$j_1 = j_2$	$1/(mr)$	
3		$\leq 2/(mr)^2$	$3 \mid m$
		0	$3 \nmid m$
4	$j_1 = j_2 = j_3 = j_4$	$\leq (1-1/m)^2/(mr^3)$	
	$j_1 = j_3, j_2 = j_4$	$(1-1/r)/(mr)^2$	

The proof of this proposition is given as an appendix. This proposition shows that when t and r are large enough, the probability to find a t-cycle is very small, and we likely do not encounter one.

Taking care of useless cycles. Using Proposition 31, it is clear that an easy way of solving the problem is to force a large value of r, which is not really a problem in practice. In the rare cases where a cycle appears, we can get out of it by *collapsing* it as suggested in [13]. If the cycle is $R_1 \mapsto R_2 \mapsto \cdots \mapsto R_t$, we want to get out of it in a *symmetric* way, that is reach the same point, whichever R_i was the point at which we entered the cycle. Our version is to sort the points R_i to obtain S_1, S_2, …, S_t and start again, say, from $R = \bigoplus_{i=1}^{t}[i^i + 1]S_i$. Anything that breaks linearity would be convenient.

From a practical point of view, we count the number of distinguished points we obtain along the path. If this number does not evolve as prescribed by the theory, that is 1 among $1/\theta$ points, then we decide to inspect the cycle and check out if it comes from a useless one or not. The more frequent case of 2-cycles can be handled with a look-ahead technique (see [52]) that is somewhat simpler.

[2] If there is one, then we have found the discrete log!

3.3 A multiplicative random walk

Let us explain the situation for ABC curves [22]. In this case, $\mathbb{K} = \mathbb{F}_{2^\ell}$ and E is defined over $\mathbb{F}_2$. Then $\lambda : (x, y) \mapsto (x^2, y^2)$ is an automorphism of order ℓ. If ℓ is odd[3], the equivalence relation is: $S \sim T \iff \exists i, S = [\pm\lambda^i]T$ with $m = 2\ell$ classes. This relation has been exploited in [13] and [52] as well as in real life computations by R. Harley (see `http://www.certicom.com/chal/`).

Gallant et al. have suggested to use a multiplicative random walk with a function f defined by $f(R) = [\mu_{\overline{R}}]R$, where $\mu_{\overline{R}}$ is a multiplier depending on the class of R. This walk is well defined, since f commutes with λ:

$$f(R') = [\mu_{\overline{R'}}]R' = [\mu_{\overline{R}}]R' = [\mu_{\overline{R}}]([\pm\lambda^i]R) = [\pm\lambda^i]([\mu_{\overline{R}}]R) \in \overline{f(R)}.$$

In order to make the method efficient, the multiples must be computed very quickly. In that case, $\mu_{\overline{R}}$ is taken to be $1 + \lambda^{\mathcal{H}(\overline{R})}$ where $\mathcal{H}$ is a hash-function sending the equivalence classes to $\{0, 1, \ldots, m-1\}$. As noted in section 2.3, the number of such multipliers should be large. For the CM examples given below this will prove not to be the case.

4 Finding new examples

Our idea is to find new examples of curves with non-trivial automorphisms. We will concentrate on algebraic curves defined over finite fields of characteristic p. We will first summarize the results concerning the number of automorphisms of (the Jacobian of) a curve.

4.1 Theoretical results

For $g = 1$, we know that we can obtain automorphisms of order 4 or 6 for some CM curves (details for both cases follow in Section 4.2) and powers of the Frobenius for ABC curves [22].

We obtain examples for $g > 1$ by considering CM hyperelliptic curves and curves defined over finite fields fixed by powers of the Frobenius, as well as curves defined over $\mathbb{Q}$ having non-trivial automorphisms.

For genus $g > 1$, we need a priori distinguish between automorphisms of a curve and automorphisms of its Jacobian. Automorphisms of a curve naturally define automorphisms on its Jacobian. By Torelli's theorem [29], all automorphisms of the Jacobian (for a fixed projective embedding and a chosen zero element) arise in this way (except for multiplication by minus one on the Jacobian of non-hyperelliptic curves). And we may identify the automorphisms of a curve and those of its Jacobian.

If $g > 1$, the number of geometric automorphisms (not considering powers of the Frobenius) does not exceed $84(g-1)$ provided that $p > g + 1$, with the exception of the curve $y^2 = x^p - x$ (see [36]). Without the restriction $p > g+1$, the

[3] If ℓ is even, we obtain a relation with ℓ classes only.

upper bound becomes $16g^4$, with again a single explicit exception: $y^q - y = x^{q+1}$ or a quotient thereof, for q a power of p (see [48]). Under the assumption that the Jacobian has maximal p-rank g, the upper bound $84g(g-1)$ holds [32].

The exceptions are of no interest to us as the Jacobians of these curves contain no large cyclic components. To some extent this occurs more generally for curves with many automorphisms. Automorphisms induce invariant subgroups in the Jacobian. When these subgroups are non-trivial for many different automorphisms, the Jacobian does not admit cyclic components of large prime order. For such a large component would itself have invariant subgroups. It follows that the selection of curves with automorphisms suitable for cryptosystems requires some care. For the decomposition of a Jacobian with given automorphism group, see [1, 16].

Examples of curves with automorphisms that have large cyclic components in their Jacobian are the curves $y^2 = x^p - x + 1$ in odd characteristic p [9]. The automorphism $x \mapsto x + 1$ of prime order p induces as invariant subgroup the trivial group and the Jacobian itself is often of prime order. Other examples are described in Section 4.3. In this paper we will always consider that the automorphism on the Jacobian can be restricted to an automorphism of the cyclic subgroup we are working in. This will be the case for all practical examples where the group is almost prime. Indeed, assume that we are dealing with a cyclic subgroup of prime cardinality n generated by P. Then n^2 does not divide the cardinality of the whole group, and the only elements of order n in the group are precisely those which constitute the subgroup $< P >$. Then the image of P by the automorphism is a of order n, and is in $< P >$, and the subgroup is stable under the action of the automorphism.

4.2 New examples with elliptic curves

We can find the proofs for what follows in [42, Chapter 2]. It is well known that the only elliptic curves having non-trivial automorphisms (over $\overline{\mathbb{Q}}$) have CM by $\mathbb{Z}[i]$ (resp. $\mathbb{Z}[\rho]$) where $i^2 = -1$ (resp. $\rho^3 = 1$). They are: $E_{a,0} : Y^2 = X^3 + aX$ for $a \neq 0$ and $E_{0,b} : Y^2 = X^3 + b$ with $b \neq 0$. On $E_{a,0}$, multiplication by i is an automorphism sending a point (x, y) to $(-x, iy)$; on $E_{0,b}$, ρ sends (x, y) to $(\rho x, y)$.

Over $\mathbb{Q}$, this is all the story, an elliptic curve having in general automorphism group $\{\pm 1\}$. Over finite fields, Frobenius automorphisms enter the game as seen with ABC curves. It is easy to see how to generalize the automorphism attack to curves suggested by Müller [30] and Smart [45].

The reductions of the curves $E_{a,0}$ and $E_{0,b}$ are supersingular for $p = 2, 3$, so that we will consider them interesting only in the case $p > 3$. For the sake of simplicity, we suppose that $\mathbb{K} = \mathbb{F}_p$, p odd prime > 3.

The case $E_{a,0}$. We suppose that $a \not\equiv 0 \bmod p$ and $p \equiv 1 \bmod 4$ (if $p \equiv 3 \bmod 4$, then $E_{a,0}$ is supersingular and the MOV reduction applies [26]). Again, we suppose that we are looking for a discrete log on the group $< P >$ with prime order n. Let $I_p^2 \equiv -1 \bmod p$ and $I_n^2 = -1 \bmod n$ such that $[I_n](x,y) = (-x, I_p y)$ (this is possible since p and n are $\equiv 1 \bmod 4$ by the theory of reduction of CM curves). Then the automorphism is $\alpha(x,y) = [I_n](x,y)$ and the class of $R = (x,y)$ consists of $\{(x,y), (-x, I_p y), (x, -y), (-x, -I_p y)\}$. We decide that $\overline{R}$ is the point with minimal ordinate in absolute value.

The "obvious" choice would be to use a generalization of the ABC approach and use $\mu_{\overline{R}} = 1 + \alpha^{\mathcal{H}(\overline{R})}$ where $\mathcal{H}$ sends the classes to $\{0,1,2,3\}$. However, this does not work, due to the fact that $\alpha^2 = -1$ and $(1+\alpha)(1-\alpha) = 2$. A random walk would compute multiples of the form $(1+\alpha)^u(1-\alpha)^v$ which is not random enough (compare with [39]). Using other multiples would be too costly.

So, we must use the additive random walk, with the modifications described in the preceding section. This gives us a speed up of $\sqrt{4} = 2$. In the case of $\mathbb{F}_{p^\ell}$ (ℓ odd), we can use the Frobenius $(x,y) \mapsto (x^p, y^p)$ to obtain a speedup of $\sqrt{4\ell}$.

The case $\boldsymbol{E_{0,b}}$. We suppose $b \not\equiv 0 \bmod p$ and $p \equiv 1 \bmod 3$ (the case $p \equiv 2 \bmod 3$ yields a supersingular curve). We let ρ_p (resp. ρ_n) denote a primitive third root of unity modulo p (resp. modulo n) such that $[\rho_n](x,y) = (\rho_p x, y)$. The automorphism is $\alpha(x,y) = [-\rho_n](x,y)$ and the class of $R = (x,y)$ consists of $\{(x, \pm y), (\rho_p x, \pm y), (\rho_p^2 x, \pm y)\}$. We take the representative with smallest x and smallest y.

Here again, the multiplicative choice using multipliers of the form $(1+\alpha^i)$ is disastrous due to the fact that $1+\alpha^2 = -\alpha$ and $1+\alpha = -\alpha^2$ and we come back to the additive version, thus yielding a speedup of $\sqrt{6}$ and more generally $\sqrt{6\ell}$ over $\mathbb{F}_{p^\ell}$, ℓ odd.

4.3 Examples of hyperelliptic curves

We will consider curves of equation $Y^2 = F(X) = X^{2g+1} + \cdots$ when $p > 2$ (resp. $Y^2 + H(X)Y = F(X) = X^{2g+1} + \cdots$ for $p = 2$) where $g > 1$ is the genus of the curve. There is no group law on the curve, but there is one on its Jacobian, noted $\mathrm{Jac}(C)$. The only result we need says that every element of $\mathrm{Jac}(C)$ can be uniquely represented by a so-called reduced divisor with at most g points. For more precise statements about those things, one can refer to Mumford [31]. In [4] (see also [20]), Cantor gives an algorithm for computing with the reduced divisors: we can add them in polynomial time. We do not recall his method here, but we give the representation of a reduced divisor: it is a pair of polynomials $[u(z), v(z)]$, where u is monic of degree at most g and greater than the degree of v. The opposite of such a reduced divisor (for the Jacobian group law) is just the divisor represented by $[u(z), -v(z)]$. The *hyperelliptic involution* is $[u(z), v(z)] \mapsto [u(z), -v(z)]$.

From a practical point of view, it is not as easy to compute the cardinality of a Jacobian over finite fields as it is for elliptic curves using the so-called SEA algorithm [25, 24]. So various researchers have suggested some special form of curves for which this computation is easy [21], [9], [5], [46], [37], [3].

Automorphisms. We suppose α is an automorphism on the curve of the form $\alpha : (x, y) \mapsto (\alpha_1(x), \alpha_2(y))$. We restrict ourself to the study of two classes of automorphisms: in case 1, the coordinate-functions α_1 and α_2 are identical and are also automorphisms (a typical example is the Frobenius action), and in case 2, α_1 and α_2 are multiplication maps in the finite field (this occurs for CM curves).

In both cases the automorphism can be extended to the Jacobian of the curve, and we still denote α the extended automorphism. With the polynomial representation of reduced divisors, we can derive simple expressions for the action of α: for $h \leq g$, we have

$$\begin{aligned} \alpha \ : \ & [z^h + u_{h-1}z^{h-1} + \cdots + u_0,\ v_{h-1}z^{h-1} + \cdots + v_0] \\ \mapsto & [z^h + \alpha_1(u_{h-1})z^{h-1} + \cdots + \alpha_1(u_0),\ \alpha_1(v_{h-1})z^{h-1} + \cdots + \alpha_1(v_0)] \end{aligned}$$

for case 1, and

$$\begin{aligned} \alpha \ : \ & [z^h + u_{h-1}z^{h-1} + \cdots + u_0,\ v_{h-1}z^{h-1} + \cdots + v_0] \\ \mapsto & [z^h + \alpha_1 u_{h-1}z^{h-1} + \cdots + \alpha_1^h u_0,\ \alpha_2\alpha_1^{-(h-1)}v_{h-1}z^{h-1} + \cdots + \alpha_2 v_0] \end{aligned}$$

for case 2.

Thus, provided that the computation of the action of α_1 and α_2 is easy we can obtain the image of a divisor quite quickly.

Examples. In the literature we find many examples of hyperelliptic curves which come with an automorphism. Some of them are obvious (such as Frobenius endomorphism), but some others were probably not known to the authors (see the curve of Sakai and Sakurai [37]). We summarize in table 1 some examples for which we give the non-trivial automorphisms, and the order m that we obtain when combining them together with the hyperelliptic involution.

Author	Equation of curve	Field	Automorphisms	m
Koblitz [20], [21]	$Y^2 + Y = X^5 + X^3$ $(*)$	$\mathbb{F}_{2^n}$	Frob + $\begin{cases} X \mapsto X+1 \\ Y \mapsto Y + X^2 \end{cases}$	$4n$
	$Y^2 + Y = X^5 + X^3 + X$ $(*)$	$\mathbb{F}_{2^n}$	Frobenius	$2n$
	$Y^2 + Y = X^{2g+1} + X$	$\mathbb{F}_{2^n}$	Frobenius	$2n$
	$Y^2 + Y = X^{2g+1}$	$\mathbb{F}_{2^n}$	Frobenius	$2n$
Buhler Koblitz [3] Chao et al. [5]	$Y^2 + Y = X^{2g+1}$ $(\dagger)$ (and twists)	$\mathbb{F}_p$ with $p \equiv 1\ (2g+1)$	mult. by ζ_{2g+1}	$2(2g+1)$
Sakai Sakurai [37]	$Y^2 + Y = X^{13} + X^{11} +$ $X^9 + X^5 + 1$ $(\dagger)$	$\mathbb{F}_{2^{29}}$	Frobenius & $\begin{cases} X \mapsto X+1 \\ Y \mapsto Y + X^6 + X^5 \\ \quad + X^4 + X^3 + X^2 \end{cases}$	4×29
Duursma & Sakurai [9]	$Y^2 = X^p - X + 1$ $(\dagger)$	$\mathbb{F}_{p^n}$	Frobenius & $(X, Y) \mapsto (X+1, Y)$	$2np$

Table 1. Examples of curves

The curves marked by $(*)$ can be broken by the reduction method of [11] which is much faster than the ρ method, even with the use of automorphisms.

For high genus curves in the examples marked by (†), a faster attack is given by the index-calculus method of [2].

For each curve in the table, we can improve the parallel collision search by a factor $\sqrt{m}$. For example, if one wants to break the cryptosystem proposed by Sakai and Sakurai, which uses a divisor of prime order around 2^{171}, we have to perform about 2^{86} operations on the Jacobian, and using the Frobenius, the involution and the new automorphism, we have only to perform about 2^{82} operations. Note that the authors took into account the use of Frobenius in their evaluation of the security, but not the other automorphism (however it still does not break the system).

The curve $Y^2 = X^5 - 1$ and generalizations. The Jacobian of the curve $Y^2 = X^5 - 1$ admits complex multiplication by the field $\mathbb{Q}(\zeta_5)$ where ζ_5 is a 5-th root of unity. Combining it with the hyperelliptic involution, we obtain an automorphism of order 10. The formulae for the action on reduced divisors are:

$$\begin{array}{rcll} \alpha & : & [z^2 + u_1 z + u_0,\ v_1 z + v_0] & \mapsto [z^2 + \zeta_5 u_1 z + \zeta_5^2 u_0,\ -\zeta_5^{-1} v_1 z - v_0] \\ & & [z + u_0, v_0] & \mapsto [z + \zeta_5 u_0, -v_0] \\ & & 0 & \mapsto 0. \end{array}$$

This case may be generalized to the curves of equations $Y^2 = X^{2g+1} - 1$ (or $Y^2 + Y = X^{2g+1}$ as suggested in [3]), which admit complex multiplication by the ring of integers of $\mathbb{Q}(\zeta_{2g+1})$, providing an automorphism of order $2g+1$ (and even $4g+2$ combined with the trivial involution).

In this case, we could also dream of using a multiplicative random walk with multipliers $1 + \alpha^i$. However, we have to be careful, due to the many algebraic relations between 5-th roots of unity. Apart from $1 - \alpha + \alpha^2 - \alpha^3 + \alpha^4 = 0$, we have among others $1 + \alpha^5 = 0$, $(1 + \alpha^2)(1 + \alpha^4) = \alpha^3$, $(1 + \alpha^4)(1 + \alpha^8) = \alpha$, $(1+\alpha^6)(1+\alpha^8) = -\alpha^2$, $(1+\alpha^3)(1+\alpha^8) = 1+\alpha$, etc. If during the random walk we do a couple of consecutive steps corresponding to one of these relations, then we obtain a cycle of length 2 which is useless. So, we come back to the additive one to get our speedup of $\sqrt{10}$.

5 Numerical experimentations

In order to validate our findings, we made several experiments on random graphs of reasonable size as well as some real discrete log computations on small examples. The hash function used was $\mathcal{H}((x, y)) = y \bmod r$ in each case. All examples were done using the computer algebra system MAGMA.

5.1 Elliptic examples

We built functional graphs for the case of $E : Y^2 = X^3 + 2X$ over $\mathbb{K} = \mathbb{F}_{1000037}$. We chose $P = (301864, 331917)$ of prime order $n = 500153$. We made two series of experiments on 50 random mappings with r branches. The following table contains the number of useless t-cycles found:

r	$\#2-\text{cycles}$	$\#4-\text{cycles}$
20	772(781)	5(5)
1000	16(15)	0(0)

These values are close to those prescribed by Proposition 31 that are given in parentheses.

We did the same for the curve $E : Y^2 = X^3 + 2$ over $\mathbb{K} = \mathbb{F}_{1000381}$ on which $P = (1, 696906)$ is a point of prime order $n = 998839$. Again we tried 50 random walks.

r	$\#2-\text{cycles}$	$\#4-\text{cycles}$	$\#3-\text{cycles}$
20	694(693)	7(7)	2(3)
1000	13(14)	0(0)	0(0)

We also did 100 real discrete log computations for the curves given below. We give the average gain obtained, that is the ratio $\theta\sqrt{\pi n/2}$ divided by the number of distinguished points computed. For $E : Y^2 = X^3 + 2X$ over $\mathbb{F}_{10000003021}$ with $P = (9166669436, 170163551)$ of prime order $n = 5000068261$, $Q = (1314815213, 7654067643)$ (with $\kappa = 2153613198$), we got a speed up of 1.88 for $r = 20$ (resp. 1.82 for $r = 1000$) and for $E : Y^2 = X^3 + 2$ over $\mathbb{F}_{10000000963}$, $P = (2, 5825971627)$, $n = 9999804109$, $Q = (4, 6715313768)$ (thus $\kappa = 8959085671$), we got 2.23 for $r = 1000$.

5.2 Hyperelliptic examples

We built the functional graphs of 50 random mappings for the Jacobian of the curve $Y^2 = X^5 - 1$ over $\mathbb{K} = \mathbb{F}_{31^3}$, which has a divisor of prime order $n = 778201$. We found 201 components on average, with 196 2-cycles and 0.8 4-cycles, which is closed to the expected theoretical values (194 and 0.7).

We did the same for the first example given by Koblitz in [20]: the curve is $Y^2 + Y = X^5 + X^3 + X$ over $\mathbb{K} = \mathbb{F}_{2^{11}}$, with a divisor of order $n = 599479$. We tried 50 different pseudo-random walks obtaining on average 31.5 2-cycles (the theory predicts 30.9), and almost no 4-cycles (0.07 expected). With the modified walk, we have 5.6 components on average (5.1 for pure random walk).

We also did some experiments of discrete log computations. For each of the curves, we did 1000 tests, the r parameter was fixed to 50. The first curve was $Y^2 = X^5 - 1$ over $\mathbb{F}_{31^7}$, with a divisor of order $n = 1440181261$. We got an average gain of 3.13 for the number of iterations (to be compared with $\sqrt{10} \approx 3.16$). The second test was done for the curve $Y^2 + Y = X^5 + X^3 + X$ over $\mathbb{F}_{2^{37}}$, with a divisor of order $n = 319020217$. The average gain obtained was 8.83 (theory says $\sqrt{74} \approx 8.60$).

6 Conclusions

We have described a general framework speeding up discrete logarithm computations on curves with large automorphism groups, including in particular some elliptic and hyperelliptic curves with complex multiplication. One could raise the question to find curves having automorphisms which are not of Frobenius or CM type.

Note that for high genus hyperelliptic curves, we can speed up the Adleman–DeMarrais–Huang discrete log algorithm [2] using all these automorphisms (see [14]). This suggests that the use of such curves in cryptosystems requires great care.

For the genus > 2, most of the curves in the literature for which the cardinality of the Jacobian can be computed have a non-trivial automorphism. It would be interesting to build new curves having no automorphisms. For genus 2, examples are given by the CM construction of Spallek [46], see also [51]. See also [47] for some examples of high genus random curves and [12] for superelliptic curves.

Acknowledgments. We are grateful to Projet CODES at INRIA, and particularly Daniel Augot, for having given the authors the opportunity to meet and work together. We also to thank E. Thomé for helpful discussions concerning this work, as well as R. Harley for his careful reading of an intermediate version of the article. Also, we are very happy to receive constructive reports from the referees, a phenomemon which is so rare that we want to emphasize it.

A Proof of Proposition 31

Remember that α is an automorphism of order m and that we use an additive function with r branches.

A.1 2-cycles

How many useless 2-cycles do we expect? We first evaluate the probability for a random point R_1 to be in a useless 2-cycle. Let R_2 be the point computed by the pseudo-random walk, and $j_1 = \mathcal{H}(R_1)$. We have $R_2 = \overline{R_1 \oplus T^{(j_1)}}$. The useless cycle can be produced when the class of $R_1 \oplus T^{(j_1)}$ is $\ominus R_1 \ominus T^{(j_1)}$, which occurs with probability $1/m$. If this first condition is satisfied, the cycle is produced when $\mathcal{H}(R_2)$ equals j_1, which occurs with probability $1/r$. Finally, the probability for a point to be in a useless 2-cycle is $1/(rm)$.

A.2 3-cycles

For 3-cycles, the expression of τ is

$$\tau = [\varepsilon_1\varepsilon_2 R_1 + \varepsilon_1\varepsilon_2]T^{(j_1)} \oplus [\varepsilon_2]T^{(j_2)} \oplus T^{(j_3)}.$$

The condition on the j_k's implies that $j_1 = j_2 = j_3$. Then $\tau = 0 = [\varepsilon_1\varepsilon_2 + \varepsilon_2 + 1]T^{(j_1)}$, and we have $\varepsilon_2(1 + \varepsilon_1) = -1$. We study "true" 3-cycles, so we do not want $R_3 = R_1$, so we can suppose $\varepsilon_1 \neq -1$. Then $\varepsilon_2 = -1/(1 + \varepsilon_1)$.

If we suppose that the automorphism α is a root of unity[4], the last equation gives $|1+\varepsilon_1| = |\varepsilon_1| = 1$, and then ε_1 is a third root of unity. Thus, if the order m of α is not a multiple of 3, we cannot have a useless 3-cycle; otherwise we can give an upper bound for the probability for a point to be in a useless 3-cycle by $2/(mr)^2$ (ε_1 can take 2 values, and then ε_2 is unique).

A.3 4-cycles

We have three different patterns for the j_k's.

First case: $j_1 = j_2 = j_3 = j_4$. Then we have $\tau = [\varepsilon_1\varepsilon_2\varepsilon_3 + \varepsilon_2\varepsilon_3 + \varepsilon_3 + 1]T^{(j_1)}$, and we have a cycle if

$$\varepsilon_3 = -\frac{1}{1+\varepsilon_2(1+\varepsilon_1)}.$$

The properties $\varepsilon_1 \neq -1$ and $\varepsilon_2 \neq -1/(1+\varepsilon_1)$ are necessary to have a true 4-cycle, and guarantee that the expression makes sense. However, for some values of ε_1 and ε_2, there is no ε_3 satisfying this equation. For example, if $\alpha^4 = 1$, then the only triples of solutions are $(\varepsilon_1, \varepsilon_2, \varepsilon_3) = (\alpha, \alpha, \alpha)$ or $(-\alpha, -\alpha, -\alpha)$.

We can bound the probability of a 4-cycle with this pattern. The probability that the j_k's are equal is $1/r^3$; the probability that $\varepsilon_1 \neq -1$ is $1 - 1/m$, the probability that $\varepsilon_2 \neq -1$ and $\varepsilon_2 \neq -1/(1+\varepsilon_1)$ is bounded by $(1 - 1/m)$, and finally we have at most one choice for ε_3, i.e. probability $1/m$. Finally, the probability for a point to belong to such a 4-cycle is less than $(1-1/m)^2/mr^3$.

Second case: $j_1 = j_2$ *and* $j_3 = j_4$. We want to have a true 4-cycle, so $R_1 \neq R_3$, which implies that $\varepsilon_1 \neq -1$. In that case, the equation $\tau = 0$ becomes

$$[\varepsilon_2\varepsilon_3(1+\varepsilon_1)]T^{(j_1)} \oplus [1+\varepsilon_3]T^{(j_3)} = 0,$$

with $T^{(j_1)} \neq T^{(j_3)}$, so this equation has no solution with $\varepsilon_1 \neq -1$ if the points $T^{(j)}$ are sufficiently randomly chosen. Hence such a pattern cannot occur with significant probability.

The same result holds for the case: $j_1 = j_4$ and $j_2 = j_3$.

Third case: $j_1 = j_3$ *and* $j_2 = j_4$. In that case, the equation $\tau = 0$ becomes $[\varepsilon_3(\varepsilon_1\varepsilon_2 + 1)]T^{(j_1)} \oplus [\varepsilon_2\varepsilon_3 + 1]T^{(j_2)} = 0$, which reduces to the system

$$\begin{cases} \varepsilon_1\varepsilon_2 = -1, \\ \varepsilon_2\varepsilon_3 = -1, \end{cases}$$

or $\varepsilon_3 = \varepsilon_1 = -1/\varepsilon_2$. If we assume that the automorphism is a root of unity, then for any given ε_1, there is a unique solution of the system for ε_2 and ε_3.

The probability that $j_1 = j_3$ is $1/r$, idem for $j_2 = j_3$, and moreover we impose $j_1 \neq j_2$, which occurs with probability $(1 - 1/r)$; the probability of finding the good values for ε_2 and ε_3 is $1/m^2$. Finally the probability for a point to belong in a 4-cycle with such a pattern is $(1-1/r)/(rm)^2$;

Putting together these results, we can bound the probability for a point to be in a useless 4-cycle by $(1-1/r)/(rm)^2 + (1-1/m)^2/mr^3$.

[4] This is always the case for an automorphism of a cyclic group.

A.4 The case of t-cycles

For a t-cycle, we rewrite the cancellation of τ as a system of k equations, one for each j_i which actually occurs. We evaluate the probability of $(\tau = 0)$ for every value of k, and then collect the results. Note that k has to be at most $t/2$, because there has to be at least two terms in each equation, and of course k is at most r. Now let $\mathcal{S}$ be the set of the j_i which actually occur in τ. Then

$$\Pr(\tau = 0) = \sum_{k=1}^{\min(r,t/2)} \Pr(\tau = 0 \,|\, \#\mathcal{S} = k).\Pr(\#\mathcal{S} = k).$$

First step: $\Pr(\#\mathcal{S} = k) \leq \min(\frac{r!k^t}{(r-k)!r^t}, 1)$.

This problem can be expressed as follows: what is the probability to get exactly k distinct elements, when one takes randomly t elements in a set of r elements (we put the element back in the set after each step). This probability is classically (see [8]) $\binom{r}{k} S_2(t,k)/r^t$, where $S_2(t,k)$ is the Stirling number of the second kind. Then we bound $S_2(t,k)$ by $k!k^t$, and obtain the formula.

Second step: $\Pr(\tau = 0 \,|\, \#\mathcal{S} = k)$.

Let us make the change of variables

$$\begin{cases} \varepsilon_1^* = \varepsilon_{t-1} \cdots \varepsilon_1 \\ \varepsilon_2^* = \varepsilon_{t-1} \cdots \varepsilon_2 \\ \quad\vdots \\ \varepsilon_{t-1}^* = \varepsilon_{t-1} \end{cases}$$

This transformation is invertible since the ε_i are invertible. The expression for τ becomes $\tau = [\varepsilon_{t-1} \cdots \varepsilon_1] T^{(j_1)} \oplus \cdots \oplus T^{(j_t)} = [\varepsilon_1^*] T^{(j_1)} \oplus \cdots \oplus [\varepsilon_{t-1}^*] T^{(j_{t-1})} \oplus T^{(j_t)} = \bigoplus_{l=1}^{k} [\mathcal{E}_l] T^{(j_{i_l})}$, where the $\mathcal{E}_l$ are *linear* expressions in the ε_i^*. When the ε_i^* are randomly chosen among m values, for each l the probability that $\mathcal{E}_l$ is verified is bounded by $1/m$. Thus the probability to have a good $(t-1)$-uple of ε_i is bounded by $1/m^k$, and the result follows.

Note that we used the fact that we have a true t-cycle, because we supposed that the ε_i were randomly chosen, which is not the case if for example we have a 2-cycle (ε_2 has an imposed value in that case).

References

1. R. D. Accola. Two theorems on Riemann surfaces with noncyclic automorphism groups. *Proc. Amer. Math. Soc.*, 25:598–602, 1970.
2. 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. 1st Algorithmic Number Theory Symposium - Cornell University, May 6-9, 1994.

3. J. Buhler and N. Koblitz. Lattice basis reduction, Jacobi sums and hyperellitic cryptosystems. *Bull. Austral. Math. Soc.*, 58:147–154, 1998.
4. D. G. Cantor. Computing in the Jacobian of an hyperelliptic curve. *Math. Comp.*, 48(177):95–101, 1987.
5. J. Chao, N. Matsuda, J. Sato, and S. Tsujii. Efficient construction of secure hyperelliptic discrete logarithm problems of large genera. In *Proc. Symposium on Cryptography and Information Security*, 1997. Fukuoka, Japan.
6. Y.-M. J. Chen. On the elliptic curve discrete logarithm problem. Preprint, June 1999.
7. J. H. Cheon, D. H. Lee, and S. G. Hahn. Elliptic curve discrete logarithms and Wieferich primes. Preprint, September 1998.
8. L. Comtet. *Analyse combinatoire.* Presses Universitaires de France, 1970.
9. I. Duursma and K. Sakurai. Efficient algorithms for the jacobian variety of hyperelliptic curves $y^2 = x^p - x + 1$ over a finite field of odd characteristic p. In H. Tapia-Recillas, editor, *Proceedings of the "International Conference on Coding Theory, Cryptography and Related Areas"*, volume yyy of *Lecture Notes in Comput. Sci.*, 1999. Guanajuato, Mexico on April, 1998.
10. P. Flajolet and A. M. Odlyzko. Random mapping statistics. In J.-J. Quisquater, editor, *Advances in Cryptology*, volume 434 of *Lecture Notes in Comput. Sci.*, pages 329–354. Springer-Verlag, 1990. Proc. Eurocrypt '89, Houthalen, April 10–13.
11. G. Frey and H.-G. Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp.*, 62(206):865–874, April 1994.
12. S. D. Galbraith, S. Paulus, and N. P. Smart. Arithmetic on superelliptic curves. Preprint, 1999.
13. R. Gallant, R. Lambert, and S. Vanstone. Improving the parallelized Pollard lambda search on binary anomalous curves. `http://www.certicom.com/chal/download/paper.ps`, 1998.
14. P. Gaudry. A variant of the Adleman-DeMarrais-Huang algorithm and its application to small genera. Research Report LIX/RR/99/04, LIX, June 1999. Available at `http://www.lix.polytechnique.fr/Labo/Pierrick.Gaudry/`.
15. M. J. Jacobson, N. Koblitz, J. H. Silverman, A. Stein, and E. Teske. Analysis of the Xedni calculus attack. Preprint, February 1999.
16. E. Kani and M. Rosen. Idempotent relations and factors of Jacobians. *Math. Ann.*, 298:307–327, 1989.
17. H. J. Kim, J. Cheon, and S. Hahn. Elliptic logarithm over a finite field and the lifting to $\mathbb{Q}$. Preprint, September 1998.
18. N. Koblitz. *A course in number theory and cryptography*, volume 114 of *Graduate Texts in Mathematics.* Springer–Verlag, 1987.
19. N. Koblitz. Elliptic curve cryptosystems. *Math. Comp.*, 48(177):203–209, January 1987.
20. N. Koblitz. Hyperelliptic cryptosystems. *J. of Cryptology*, 1:139–150, 1989.
21. N. Koblitz. A family of jacobians suitable for discrete log cryptosystems. In S. Goldwasser, editor, *Advances in Cryptology – CRYPTO '88*, volume 403 of *Lecture Notes in Comput. Sci.*, pages 94–99. Springer–Verlag, 1990. Proceedings of a conference on the theory and application of cryptography held at the University of California, Santa Barbara, August 21-25, 1988.
22. N. Koblitz. CM-curves with good cryptographic properties. In Joan Feigenbaum, editor, *Advances in Cryptology – CRYPTO '91*, volume 576 of *Lecture Notes in Comput. Sci.*, pages 279–287. Springer-Verlag, 1992. Santa Barbara, August 12–15.

23. H. W. Lenstra, Jr. Factoring integers with elliptic curves. *Ann. of Math. (2)*, 126:649–673, 1987.
24. R. Lercier. Finding good random elliptic curves for cryptosystems defined over F_{2^n}. In W. Fumy, editor, *Advances in Cryptology – EUROCRYPT '97*, volume 1233 of *Lecture Notes in Comput. Sci.*, pages 379–392. Springer-Verlag, 1997. International Conference on the Theory and Application of Cryptographic Techniques, Konstanz, Germany, May 1997, Proceedings.
25. R. Lercier and F. Morain. Counting the number of points on elliptic curves over finite fields: strategies and performances. In L. C. Guillou and J.-J. Quisquater, editors, *Advances in Cryptology – EUROCRYPT '95*, volume 921 of *Lecture Notes in Comput. Sci.*, pages 79–94, 1995. International Conference on the Theory and Application of Cryptographic Techniques, Saint-Malo, France, May 1995, Proceedings.
26. A. Menezes, T. Okamoto, and S. A. Vanstone. Reducing elliptic curves logarithms to logarithms in a finite field. *IEEE Trans. Inform. Theory*, IT–39(5):1639–1646, September 1993.
27. A. J. Menezes. *Elliptic curve public key cryptosystems.* Kluwer Academic Publishers, 1993.
28. 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. Proceedings, Santa Barbara (USA), August 11–15, 1986.
29. J. S. Milne. Jacobian varieties. In G. Cornell and J. H. Silverman, editors, *Arithmetic Geometry*, pages 167–212. Springer-Verlag, 1986.
30. V. Müller. Fast multiplication on elliptic curves over small fields of characteristic two. *J. of Cryptology*, 11(4):219–234, 1998.
31. D. Mumford. *Tata lectures on theta II.* Birkhauser, 1984.
32. S. Nakajima. p-ranks and automorphism groups of algebraic curves. *Trans. Amer. Math. Soc.*, 303(2):595–607, October 1987.
33. S. Pohlig and M. Hellman. An improved algorithm for computing logarithms over $\mathrm{GF}(p)$ and its cryptographic significance. *IEEE Trans. Inform. Theory*, IT–24:106–110, 1978.
34. J. M. Pollard. Monte Carlo methods for index computation $\pmod p$. *Math. Comp.*, 32(143):918–924, July 1978.
35. J.-J. Quisquater and J.-P. Delescaille. How easy is collision search? application to DES. In J.-J. Quisquater, editor, *Advances in Cryptology*, volume 434 of *Lecture Notes in Comput. Sci.*, pages 429–434. Springer-Verlag, 1990. Proc. Eurocrypt '89, Houthalen, April 10–13.
36. P. Roquette. Abschätzung der Automorphismenanzahl von Funktionenkörpern bei Primzahlcharakteristik. *Math. Z.*, 117:157–163, 1970.
37. Y. Sakai and K. Sakurai. Design of hyperelliptic cryptosystems in small charatcteristic and a software implementation over $\mathbb{F}_{2^n}$. In K. Ohta and D. Pei, editors, *Advances in Cryptology*, volume 1514 of *Lecture Notes in Comput. Sci.*, pages 80–94. Springer-Verlag, 1998. Proc. Asiacrypt '98, Beijing, October, 1998.
38. 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.
39. J. Sattler and C. P. Schnorr. Generating random walks in groups. *Ann. Univ. Sci. Budapest. Sect. Comput.*, 6:65–79, 1985.
40. 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.

41. D. Shanks. Class number, a theory of factorization, and genera. In *Proc. Symp. Pure Math. vol. 20*, pages 415–440. AMS, 1971.
42. J. H. Silverman. *Advanced Topics in the Arithmetic of Elliptic Curves*, volume 151 of *Grad. Texts in Math.* Springer-Verlag, 1994.
43. J. H. Silverman. The XEDNI calculus and the elliptic curve discrete logarithm problem. Preprint, August 1998.
44. N. Smart. The discrete logarithm problem on elliptic curves of trace one. Preprint HP-LABS Technical Report (Number HPL-97-128). To appear in *J. Cryptology*, 1997.
45. N. P. Smart. Elliptic curve cryptosystems over small fields of odd characteristic. *J. of Cryptology*, 12(2):141–151, 1999.
46. A.-M. Spallek. *Kurven vom Geschlecht 2 und ihre Anwendung in Public-Key-Kryptosystemen*. PhD thesis, Universität Gesamthochschule Essen, July 1994.
47. A. Stein and E. Teske. Catching kangaroos in function fields. Preprint, March 1999.
48. H. Stichtenoth. Über die Automorphismengruppe eines algebraischen Funktionenkörpers von Primzahlcharakteristik. I. Eine Abschätzung der Ordnung der Automorphismengruppe. *Arch. Math. (Basel)*, 24:527–544, 1973.
49. E. Teske. Speeding up Pollard's rho method for computing discrete logarithms. In J. P. Buhler, editor, *Algorithmic Number Theory*, volume 1423 of *Lecture Notes in Comput. Sci.*, pages 541–554. Springer-Verlag, 1998. Third International Symposium, ANTS-III, Portland, Oregon, june 1998, Proceedings.
50. P. C. van Oorschot and M. J. Wiener. Parallel collision search with cryptanalytic applications. *J. of Cryptology*, 12:1–28, 1999.
51. P. van Wamelen. Examples of genus two CM curves defined over the rationals. *Math. Comp.*, 68(225):307–320, January 1999.
52. M. J. Wiener and R. J. Zuccherato. Faster attacks on elliptic curve cryptosystems. In S. Tavares and H. Meijer, editors, *Selected Areas in Cryptography '98*, volume 1556 of *Lecture Notes in Comput. Sci.*. Springer-Verlag, 1999. 5th Annual International Workshop, SAC'98, Kingston, Ontario, Canada, August 17-18, 1998, Proceedings.