Computing the number of points of elliptic curves over finite fields

Johannes Buchmann and Volker Müller
FB 14 Informatik
Universität des Saarlandes
D-6600 Saarbrücken
Germany
email: vmueller@cs.uni-sb.de

1 Introduction

In this paper we study the problem of counting the points on an elliptic curve over a prime field.
This problem is not only very interesting for number theorists but it has recently gained a lot of attention among cryptographers. The use of elliptic curves in public key cryptography was suggested by Koblitz ([Ko86]) and Miller ([Mi86]). The security of their elliptic curve cryptosystems is based on the intractability of the problem of computing discrete logarithms in the elliptic curve group. The best algorithms known for solving this problem for arbitrary elliptic curves are the exponential square root attacks ([Od86]) which have running time proportional to the largest prime factor dividing the group order. Consequently, in order to guarantee the security of the system it is neccessary to find this group order and its prime factorization. Since the size of the groups used in practical applications is of order of magnitude 10^{40}, factoring the group order is easy. So it remains to find efficient algorithms for determining that order.

Although Schoof ([Scho85]) proves that the cardinality of an elliptic curve group over a finite field can be computed in polynomial time, his algorithm is extremely inefficient in practice.
On the other hand, the application of Shanks' babystep giantstep idea ([Sh70]) to our problem yields an algorithm which is efficient for medium size prime numbers but of exponential complexity.

So far no experimental results concerning those algorithms have been published. In this paper we present a practical improvement of the algorithm of Shanks which is based on the ideas of Schoof. It turns out to be very efficient as can be seen from the tables at the end of the paper.

2 Introduction to Elliptic Curves

Let F_q be a finite field with q elements of characteristic p not equal to 2 or 3. Let E be an elliptic curve defined by the affine equation

$$y^2 = x^3 + ax + b$$

with $a, b \in F_q$ and $4a^3 + 27b^2 \neq 0$.

Then the group of points of this elliptic curve is

$$E(F_q) = \left\{ (x, y) \in F_q^2 \; ; \; y^2 = x^3 + ax + b \right\} \bigcup \left\{ O \right\}. \tag{1}$$

There is a natural addition defined on the points of $E(F_q)$ that is given by the "tangent and chord method". Under this addition, the points of $E(F_q)$ form an abelian group ([Si86] or [Hu87]). This addition is defined as follows:

First the identity element of the group is O. For an ordinary point (x, y) one defines $-(x, y) = (x, -y)$. For two points (x_1, y_1) and (x_2, y_2) which are not inverse to each other $(x_1, y_1) + (x_2, y_2) = (x_3, y_3)$ where

$$x_3 = -x_1 - x_2 + \lambda^2$$

$$y_3 = -y_1 + \lambda(x_1 - x_3)$$

and λ is given by

$$\lambda = \frac{y_1 - y_2}{x_1 - x_2} \qquad \text{if } x_1 \neq x_2$$

and

$$\lambda = \frac{3x_1^2 + a}{2y_1} \qquad \text{if } x_1 = x_2.$$

A very important fact concerning the group order is given by Hasse's theorem (see [Si86]).

2.1 Theorem *Let F_q be the field with q elements and E be an elliptic curve defined over F_q. Then*

$$\mid \#E(F_q) - q - 1 \mid \leq 2\sqrt{q}. \tag{2}$$

3 The Babystep - Giantstep Strategy

In this section we describe Shanks' algorithm for computing the order of the group of points of an elliptic curve defined over the prime field F_p.

3.1 The Algorithm

Step 1 Choose a random point P in $E(F_p)$, set $k = \lceil 2\sqrt[4]{p} \rceil$.

Step 2 Compute $i \cdot P$ for $i = 0, \dots, k-1$.
If for some i we have $i \cdot P = O$ then go back to step 1.

Otherwise we store $(i \cdot P, i)$ in a table.

Step 3 Set $Q = k \cdot P$.

Step 4 Compute

$$H_j = \lfloor p + 1 - 2\sqrt{p} \rfloor \cdot P + j \cdot Q$$

for $j = 1, \dots, k$ and check whether $H_j = i \cdot P$ for some i.

If we find only one pair (i, j) with $H_j = i \cdot P$ then we have

$$\# E(F_p) = \lfloor p + 1 - 2\sqrt{p} \rfloor + k \cdot j - i$$

and the algorithm terminates.

Otherwise we find at least two successively stored pairs (i, j) and (i', j') and we can compute the order of the point P by

$$l = \mathrm{ord}(P) = |k(j - j') - (i - i')|.$$

If $l < \sqrt{p} - 1$ then go back to step 1.

Step 5 Randomly choose another point P' in $E(F_p)$.

Step 6 Compute the order l' of the point P' (analogous to steps 2 - 4).

Find the order of P' in $E(F_p)/{<}P{>}$, i.e. the least divisor d' of l' such that $d' \cdot P' \in {<}P{>}$. This can be done by a babystep giantstep method.

If $d = l \cdot d' < 4\sqrt{p}$, go back to step 5.

Step 8 Determine the unique multiple x of d in $[\, p + 1 - 2\sqrt{p}\ ,\ p + 1 + 2\sqrt{p}\,]$. Then $\# E(F_p) = x$ and the algorithm terminates.

We sketch a proof of the validity of the algorithm. The terminating condition in step 4 is correct since by Theorem 2.1 the group order of $E(F_p)$ is in the interval $[\, p + 1 - 2\sqrt{p}\ ,\ p + 1 + 2\sqrt{p}\,]$. On the other hand, the group order is a multiple of the element order of P and if there is exactly one multiple of the order of P in that interval, i.e. if $|{<}P{>}| \geq 4\sqrt{p}$, then it must be the group order.

Now it is well known that the structure of the elliptic curve group is

$$E(F_p) \quad \cong \quad Z_{/m} \times Z_{/n}.$$

Without loss of generality we assume that $m \geq n$. If $m \geq 4\sqrt{p}$ then it can be shown that a large fraction of points on E has order $\geq 4\sqrt{p}$. If such a point is found then the algorithm terminates already in step 4.

In any case, it follows from Theorem 2.1 that $|E(F_p)| \geq (\sqrt{p}-1)^2$ and therefore $m \geq \sqrt{p}-1$. Thus we are very likely to find in step 1 a point of order $\geq \sqrt{p}-1$ after polynomially many iterations of steps 1 to 4. In order to complete the computation it is now sufficient to find a point P' which has order ≥ 5 in $E(F_p)/<P>$. But if $m < 4\sqrt{p}$ then it is very likely that such a point is quickly found.

For a random curve the algorithm will always terminate in step 4. But there are, for example, curves with $p = k^2 + 1$ and $E(F_p) \cong Z_{/k} \times Z_{/k}$. For those curves we need the additional steps.

3.2 Our Implementation

We have implemented this algorithm using the computer algebra system PARI (for a description see [BBCO90]).

A random point on E is found in the following way:

Choose random elements $x \in F_p$ and compute $x^3 + ax + b$ until

$$\left(\frac{x^3 + ax + b}{p}\right) \quad = \quad 1$$

where $\left(\frac{a}{b}\right)$ denotes the Legendre symbol. Compute a square root y of $x^3 + ax + b \bmod p$ using the RESSOL-algorithm ([Sh72]). Then $P = (x, y)$ is a point of $E(F_p)$.

To minimize the size of the table we store only the x-coordinates of the points $i \cdot P$. Since a point $-Q$ can easily be computed from the point Q, we only need to store the points $i \cdot P$ for $i = 0, \ldots, \lceil \frac{k}{2} \rceil$ in step 2. This leads to an obvious modification of step 4.

The table is stored on the internal stack of PARI. After the babysteps we use quicksort (see [Me84]) to sort the elements of the table. Thus we can use binary search in step 4 of the algorithm to check whether there exists a table entry $(i \cdot P, i)$ with $H_j = i \cdot P$.

4 Schoof's Algorithm

Let $\overline{F_p}$ denote the algebraic closure of F_p.

We define an endomorphism on $E(\overline{F_p})$ by

$$\Phi : \quad (x, y) \longrightarrow (x^p, y^p). \tag{3}$$

This endomorphism is called the *Frobenius endomorphism.*

By Hasses theorem we know that $\# E(F_p) = p + 1 - c$ for a $|c| \leq 2\sqrt{p}$. Furthermore it can be proven that the Frobenius endomorphism fulfills the relation

$$\Phi^2 - c\Phi + p = 0. \tag{4}$$

Thus we can compute the group order if we can compute the trace c of the Frobenius endomorphism.

For this purpose we introduce the *division polynomials* $\psi_n(x, y) \in F_p[x, y]$. These polynomials can be computed by recursive equations ([Scho85]).

The division polynomials have the following useful properties:

4.1 Theorem *Let $P = (x, y) \in E(\overline{F_p})$ with $2 \cdot P \neq O$; let $n \in Z, n \geq -1$. Then*

$$n \cdot P = O \quad \Longleftrightarrow \quad \psi_n(x, y) = 0 \tag{5}$$

and if $n \geq 1$ and $n \cdot P = (x', y') \neq O$ we have

$$x' = x - \frac{\psi_{n-1}\psi_{n+1}}{\psi_n^2} \tag{6}$$

and

$$y' = \frac{\psi_{n+2}\psi_{n-1}^2 - \psi_{n-2}\psi_{n+1}^2}{4y\psi_n^3}. \tag{7}$$

(By ψ_k we mean $\psi_k(x, y)$).

A proof of this theorem can be found in [La78].

These formulas will enable us to compute on l-torsion points of $E(\overline{F_p})$ for $l \geq 3$.

Furthermore we can compute the trace c of the Frobenius equation modulo l by restricting the Frobenius endomorphism to the subgroup of the l-torsion points. Note that – if p does not divide l – this subgroup is different from zero. We simply have to check for which $\tau \in \{0, \ldots, l-1\}$ the relation $\Phi^2 + p = \tau\Phi$ holds for all l-torsion points. According to the last theorem this test can be accomplished by computations using polynomials in $F_p[x, y]$. The exact polynomial equations can be found in [Scho85].

It should be noted that the determination of $c \bmod 2$ is easy:

4.1 Lemma *Let $P = (x, y) \in E(\overline{F_p})$. Then we have $2 \cdot P = O$ if and only if $y = 0$.*

Thus $c \bmod 2$ can be determined by checking whether $x^3 + ax + b$ has a zero in F_p.

If we know $c \bmod l$ for "sufficiently many" l, we can compute c (and so the group order) by using the chinese remainder theorem.

For more details see the original paper of Schoof ([Scho85]) and [ChRo88].

5 Combination of Schoof's and Shanks' Idea

The main problem of Schoof's algorithm is the fact that the degree of the division polynomials is approximately $\deg(\psi_n) \approx \frac{n^2}{2}$. Therefore the computation of $c \bmod l$ in reasonable time is only possible for small l.

In our algorithm we first compute $\#E(F_p) \bmod l_1 \ldots l_r$ for a few small primes $l_1, \ldots, l_r$ using Schoof's idea. Then we use the babystep-giantstep algorithm in a slightly different version: in step 1 we choose

$$k = \min\{k' \mid k' \geq \lceil 2\sqrt[4]{p}\,\rceil,\ k' \equiv 0 \bmod l_1 \ldots l_r\}.$$

Thus in step 2 it is sufficient to store $i \cdot P$ only for i with

$$i \equiv (\ \lfloor p + 1 - 2\sqrt{p}\,\rfloor - \#E(F_p)\ ) \bmod l_1 \ldots l_r.$$

Using this fact the size of the table can be reduced by a factor of $\frac{1}{l_1 \ldots l_r}$.

6 Practical Results

We present some running time results for the babystep-giantstep algorithm and our combined algorithm.

It can be proven that the expected running time of Shanks' algorithm is $O(p^{\frac{1}{4}+\epsilon})$. We present the following results for the practical running time (all computations were done on a SUN 1 SPARC-station with 16 MBYTE main memory).

size of prime	running time
$\approx 10^{11}$	7 sec
$\approx 10^{12}$	11 sec
$\approx 10^{13}$	24 sec
$\approx 10^{14}$	44 sec
$\approx 10^{15}$	1 min 28 sec
$\approx 10^{16}$	3 min 2 sec
$\approx 10^{17}$	5 min 53 sec
$\approx 10^{18}$	9 min 8 sec
$\approx 10^{19}$	20 min 55 sec
$\approx 10^{20}$	34 min 27 sec
$\approx 10^{21}$	54 min 25 sec
$\approx 10^{22}$	1 h 52 min 53 sec
$\approx 10^{23}$	3 h 54 min 23 sec
$\approx 10^{24}$	8 h 44 min 27 sec

Next we present the running time results of the combined Schoof-Shanks algorithm for the same elliptic curves. For these results we used only the small primes 2, 3, 5 and 7 in the Schoof part.

size of prime	running time
$\approx 10^{11}$	1 min 39 sec
$\approx 10^{12}$	1 min 25 sec
$\approx 10^{13}$	2 min 11 sec
$\approx 10^{14}$	1 min 34 sec
$\approx 10^{15}$	2 min 39 sec
$\approx 10^{16}$	1 min 9 sec
$\approx 10^{17}$	3 min 13 sec
$\approx 10^{18}$	2 min 57 sec
$\approx 10^{19}$	3 min 46 sec
$\approx 10^{20}$	4 min 48 sec
$\approx 10^{21}$	8 min 55 sec
$\approx 10^{22}$	14 min 25 sec
$\approx 10^{23}$	24 min 11 sec
$\approx 10^{24}$	34 min 39 sec
$\approx 10^{25}$	1 h 8 min 54 sec
$\approx 10^{26}$	2 h 25 min 38 sec
$\approx 10^{27}$	4 h 28 min 58 sec

In all these examples the time needed for the Schoof part was less than 4 minutes.

7 Possible Improvements

We are currently working on the following improvements:

- Allowing more primes in the Schoof part of the algorithm.
- Using modified division polynomials as suggested by Charlap, Coley and Robbins [CCRo90].

We are also working on a complexity analysis of our method.

Literatur

[ChRo88] L. Charlap, D. Robbins: *An elementary introduction to elliptic curves*, CRD Expository Report No. 31, Institude for Defense Analyses Princeton, December 1988

[CCRo90] L. Charlap, R. Coley, D. Robbins: *Enumeration of Rational points on elliptic curves over finite fields*, preprint

[Hu87] D. Husemöller: *Elliptic curves*, Graduate texts in Mathematics 111, Springer-Verlag, 1987

[Ko86] N. Koblitz: *Elliptic curve cryptosystems*, Mathematics of Computation, 48 (1987), 203-209

[La78] S. Lang: *Elliptic Curves, Diophantine Analysis*, Springer-Verlag, 1978

[Li87] J. van der Lingen: *Elliptic curves and factorization* , Report 87-2 (1987), University of Amsterdam

[Me84] K. Mehlhorn: *Data Structures and Algorithms1: Sorting and Searching*, Springer-Verlag, 1984

[Mi86] V. Miller: *Uses of elliptic curves in cryptography*, Advances in Cryptology: Proceedings of Crypto '85, Lecture Notes in Computer Science, 218 (1986), Springer-Verlag, 417-426

[Od86] A. Odlyzko: *Discrete logarithms and their cryptographic significance*, Advances in Cryptology: Proceedings of Eurocrypt '84, Lecture Notes in Computer Science, 209 (1985), Springer-Verlag, 224-314

[BBCO90] C. Batut, D. Bernardi, H. Cohen, M. Olivier: *User's guide to PARI -GP*

[Scho85] R. Schoof: *Elliptic curves over finite fields and the computation of square roots mod p*, Mathematics of Computation, vol. 44 , 1985, 483-494

[Sh70] D. Shanks: *Class number, A theory of factorization and genera*, Proc. Sympos. Pure Math. vol.20, Amer. Math. Soc. , 1970, 415-440

[Sh72] D. Shanks: *Five number theoretic algorithms*, Congressus Numerantium No. VII, Proc. 2nd Manitoba Conf. on Numerical Math. , 1972, 51-70

[Si86] J. Silverman: *The arithmetic of Elliptic Curves*, Springer-Verlag, 1986