BULL. AUSTRAL. MATH. SOC. 11Y16
VOL. 72 (2005) [251–263]

FINDING THE GROUP STRUCTURE OF ELLIPTIC CURVES OVER FINITE FIELDS

JOHN B. FRIEDLANDER, CARL POMERANCE AND IGOR E. SHPARLINSKI

We show that an algorithm of V. Miller to compute the group structure of an elliptic curve over a prime finite field runs in probabilistic polynomial time for almost all curves over the field. Important to our proof are estimates for some divisor sums.

1. INTRODUCTION

Let $p > 3$ be prime and let $\mathbb{F}_p$ denote the field of p elements. Let $\mathbf{E}$ be an elliptic curve over $\mathbb{F}_p$ given by an affine *Weierstrass equation* of the form

$$y^2 = x^3 + ax + b, \tag{1}$$

with coefficients $a, b \in \mathbb{F}_p$, such that $4a^3 + 27b^2 \neq 0$. In particular, there are $p^2 + O(p)$ distinct elliptic curves over $\mathbb{F}_p$.

We recall that the set $\mathbf{E}(\mathbb{F}_p)$ of $\mathbb{F}_p$-rational points on any elliptic curve $\mathbf{E}$ forms an Abelian group (with a point at infinity as the neutral element) and the cardinality of this group satisfies the *Hasse–Weil* bound

$$\left|\#\mathbf{E}(\mathbb{F}_p) - p - 1\right| \leqslant 2p^{1/2}, \tag{2}$$

see [**4, 6, 22**] for this and some other general properties of elliptic curves.

It is also well known, see [**4, 6, 22**], that the group of $\mathbb{F}_p$-rational points $\mathbf{E}(\mathbb{F}_p)$ is isomorphic to

$$\mathbf{E}(\mathbb{F}_p) \cong \mathbb{Z}_M \times \mathbb{Z}_L \tag{3}$$

for unique integers M and L with

$$L \mid M \qquad \text{and} \qquad L \mid p - 1. \tag{4}$$

Received 13th April, 2005
During the preparation of this paper, the first author was supported in part by NSERC grant A5123 and by a Killam Research Fellowship. The second author was supported in part by NSF grant DMS-0401422 and the third author was supported in part by ARC grant DP0211459.

In particular, $L \mid D_p(\mathbf{E})$ where

$$D_p(\mathbf{E}) = \gcd(\#\mathbf{E}(\mathbb{F}_p), p-1).$$

The algorithm of Schoof [**19**] computes $\#\mathbf{E}(\mathbb{F}_p)$ in deterministic polynomial time, see also [**2, 6**] for more recent improvements (both theoretical and practical). However, computing the group structure (3) seems to be harder.

We recall that the *exponent* of a finite Abelian group is the largest possible order of an element. In view of (3) and (4) the exponent of $\mathbf{E}(\mathbb{F}_p)$ is just M. Since $\#\mathbf{E}(\mathbb{F}_p) = ML$, once $\#\mathbf{E}(\mathbb{F}_p)$ is known finding the group structure is equivalent to finding M or L.

The deterministic algorithm of [**9**] for computing the group structure runs in exponential time $p^{1/2+o(1)}$. We are concerned here mainly with the probabilistic algorithm of Miller [**16**]. This algorithm uses the Weil pairing to produce a pair of generators of $\mathbf{E}(\mathbb{F}_p)$.

Miller's algorithm runs in expected polynomial time plus the time needed to factor $D_p(\mathbf{E})$. So an interesting task is to analyse how likely it is for $D_p(\mathbf{E})$ to be small enough to guarantee that it can be factored in polynomial time.

Here we show that even rather slow deterministic factoring algorithms are already good enough to factor $D_p(\mathbf{E})$ in deterministic polynomial time "on average" over all curves $\mathbf{E}$ over $\mathbb{F}_p$. Then we use more advanced, but probabilistic, factoring algorithms and obtain better bounds. In fact these bounds show that seemingly the most time–consuming part of the whole algorithm, that is factoring $D_p(\mathbf{E})$, is "on average" easier than the other parts, for example the computation of $\#\mathbf{E}(\mathbb{F}_p)$. We also consider still faster heuristic algorithms and analyse the average complexity of computing $D_p(\mathbf{E})$ under the standard assumptions about the complexity of these algorithms. Under those assumptions we prove even tighter bounds.

In the opposite direction we show that there are infinitely many pairs p, $\mathbf{E}$ for which $D_p(\mathbf{E})$ is a product of two large primes and hence is probably hard to factor. Thus, our results concerning average complexity cannot be extended to the worst case complexity unless a polynomial time factoring algorithm is found.

Our estimates are based on bounds for certain divisor sums. As usual we use $\tau(k)$ and $\omega(k)$ to denote the number of of positive integer divisors and the number of prime divisors of an integer $k \geqslant 1$. We use the well-known bounds

$$\tau(k) = k^{o(1)}, \tag{5}$$

see [**18**, Theorem 5.2 of Chapter 1], and

$$\omega(k) \ll \frac{\log k}{\log\log(k+2)}. \tag{6}$$

which follows, for example, from the inequality $\omega(k)! \leqslant k$ and Stirling's formula.

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

Throughout the paper we use p and ℓ to denote prime numbers. All implied constants in the symbols 'O' and '$\ll$' are absolute (recall that $A \ll B$ is equivalent to $A = O(B)$), and $\log z$ denotes the natural logarithm of $z > 0$.

2. Preliminaries

We start with the observation that the Miller algorithm [**16**], whether it uses a deterministic factoring algorithm or a probabilistic factoring algorithm complemented by the polynomial time primality test [**1, 13**], is of *Las Vegas* type. That is, it may occasionally run in exponential time (or never terminate), but it never gives a wrong answer. Moreover, for any elliptic curve $\mathbf{E}$ given by the Weierstrass equation (1) over $\mathbb{F}_p$, if it terminates the algorithm returns the exponent of the group $\mathbf{E}(\mathbb{F}_p)$ plus a deterministic polynomial time certificate for this exponent. In fact, it also produces a set of generators of $\mathbf{E}(\mathbb{F}_p)$.

As usual we say that an integer d is *Q-smooth* if all prime divisors $\ell \mid d$ satisfy $\ell \leqslant Q$. We also say that an integer d is *Q-rough* if all prime divisors $\ell \mid d$ satisfy $\ell > Q$.

We now outline our approach to the factorisation of $D_p(\mathbf{E})$. In fact, we follow the standard three-stage strategy:

Stage 1. Set a certain smoothness bound Q, find all prime divisors $\ell \mid D_p(\mathbf{E})$ with $\ell \leqslant Q$ together with the exact powers in which they divide $D_p(\mathbf{E})$ and compute the largest Q-rough divisor $D_{Q,p}(\mathbf{E}) \mid D_p(\mathbf{E})$.

Stage 2. Test $D_{Q,p}(\mathbf{E})$ for primality.

Stage 3. If $D_{Q,p}(\mathbf{E})$ is not prime, factor $D_{Q,p}(\mathbf{E})$.

Because we are mainly interested in asymptotic results we always assume that fast arithmetic is used, thus any arithmetic operation with two b-bit integer operands can be performed in $b^{1+o(1)}$ bit operations (which is also the measure of the algorithm run time).

Accordingly, for Stage 1 we consider the following algorithms to find Q-smooth and Q-rough parts of an integer $n \geqslant 1$:

1. Deterministic Pollard–Strassen smoothness test which runs in time $\mathcal{S}(Q, n) = Q^{1/2}(\log n)^{1+o(1)}$, see pages 107–108 of [**17**].
2. Probabilistic Lenstra–Pila–Pomerance hyperelliptic smoothness test which runs in expected time $\mathcal{S}(Q, n) = \exp\left(c_0(\log Q)^{2/3}(\log\log Q)^{1/3}\right)(\log n)^{1+o(1)}$ for some positive constant c_0, see [**11**].
3. Lenstra elliptic curve smoothness test which is conjectured to run in expected time $\mathcal{S}(Q, n) = \exp\left(\sqrt{\left(2 + o(1)\right)\log Q \log\log Q}\right)(\log n)^{1+o(1)}$, see [**10**].

For Stage 2 we always apply the deterministic polynomial time algorithm of [**13**] which tests an integer $n \geqslant 1$ as to whether it is a prime (or prime power) in time

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

$\mathcal{P}(n) = (\log n)^{6+o(1)}$. There are faster probabilistic algorithms but they do not lead to any substantial improvement of our result (although certainly they could be of great practical value).

It is useful to remark that the above smoothness tests can also be used as factoring algorithms whose running time becomes $S\Big(\min\big\{P(n), n^{1/2}\big\}, n\Big)$ where $P(n)$ is the largest prime factor of n.

Accordingly, for Stage 3 we consider the following algorithms to factor an integer $n \geqslant 1$:

1. Deterministic Pollard–Strassen factorisation algorithm which runs in time $\mathcal{F}(n) = \min\big\{P(n)^{1/2}, n^{1/4}\big\}(\log n)^{1+o(1)}$;
2. Probabilistic Lenstra–Pomerance factorisation algorithm which runs in expected time $\mathcal{F}(n) = \exp\Big(\big(1+o(1)\big)\sqrt{\log n \log\log n}\Big)$, see [**12**];
3. Heuristic number field factorisation algorithm which is predicted to run in time $\mathcal{F}(n) = \exp\Big(\big(64/9+o(1)\big)(\log n)^{1/3}(\log\log n)^{2/3}\Big)$, see [**6**, Section 6.2]. (The algorithm of Coppersmith [**5**] replaces "64/9" with a slightly smaller number.)

So, setting the smoothness bound Q and using a smoothness algorithm whose run time is $\mathcal{S}(Q,n)$ together with a factorisation algorithm whose run time is $\mathcal{F}(n)$, we see that $D_p(\mathbf{E})$ can be factored in time

$$\mathcal{S}\big(Q, D_p(\mathbf{E})\big) + \mathcal{P}\big(D_{Q,p}(\mathbf{E})\big) + \mathcal{F}\big(D_{Q,p}(\mathbf{E})\big) \leqslant \mathcal{S}(Q,p-1) + \mathcal{P}\big(D_{Q,p}(\mathbf{E})\big) + \mathcal{F}\left(D_{Q,p}(\mathbf{E})\right).$$

Therefore, to estimate the average time for a given choice of Q and the corresponding algorithms and then optimise the choice of Q, we need some results about the distribution of $D_{Q,p}(\mathbf{E})$.

From [**10**, Proposition 1.9] (and the counting of isomorphisms and automorphisms of elliptic curves summarised in [**10**, Section 1]) we have the following result.

LEMMA 1. *For any N, the number of elliptic curves $\mathbf{E}$ given by (1) with $a, b \in \mathbb{F}_p$ and such that $\#\mathbf{E}_{a,b}(\mathbb{F}_p) = N$ is at most $O\left(p^{3/2}\log p(\log\log p)^2\right)$.*

We are now ready to establish a general upper bound on the average time complexity of factoring $D_p(\mathbf{E})$.

LEMMA 2. *Given a smoothness bound Q and a choice of a smoothness algorithm whose run time is $\mathcal{S}(Q,n)$, a primality test whose run time is $\mathcal{P}(n)$ and a factorisation algorithm whose run time is $\mathcal{F}(n)$, the average time complexity of factorisation of $D_p(\mathbf{E})$*

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

over all elliptic curves $\mathbf{E}$ *given by* (1) *with* $a, b \in \mathbb{F}_p$ *is*

$$T(p) \ll S(Q, p-1) + \log p (\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \mathcal{P}(d) \left(d^{-1} + p^{-1/2}\right) + \log p (\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \omega(d) \geqslant 2 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \mathcal{F}(d) \left(d^{-1} + p^{-1/2}\right).$$

PROOF: Using (2) and Lemma 1 we obtain

$$\begin{aligned} T(p) &\ll S(Q, p-1) + p^{-1/2} \log p (\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \mathcal{P}(d) \sum_{\substack{|N-p-1| \leqslant 2p^{1/2} \\ d \mid N}} 1 \\ &\quad + p^{-1/2} \log p (\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \omega(d) \geqslant 2 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \mathcal{F}(d) \sum_{\substack{|N-p-1| \leqslant 2p^{1/2} \\ d \mid N}} 1 \\ &\ll S(Q, p-1) + p^{-1/2} \log p (\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \mathcal{P}(d) \left(\frac{4p^{1/2}}{d} + 1\right) \\ &\quad + p^{-1/2} \log p (\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \omega(d) \geqslant 2 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \mathcal{F}(d) \left(\frac{4p^{1/2}}{d} + 1\right), \end{aligned}$$

which finishes the proof. □

Thus, to get specific upper bounds for concrete smoothness and factoring algorithms with an optimally chosen smoothness bound Q, we need to estimate certain sums over divisors.

Clearly, if the average time to factor $D_p(\mathbf{E})$ is polynomial, say $(\log p)^{A+o(1)}$ then it remains polynomial for almost all elliptic curves $\mathbf{E}$ over $\mathbb{F}_p$. Namely it may exceed $(\log p)^{A+1}$ for at most $p^2(\log p)^{-1+o(1)}$ curves.

Finally, it is also interesting to study, for a given smoothness bound Q, the number of elliptic curves $\mathbf{E}$ over $\mathbb{F}_p$, for which the second stage is needed at all, that is, for how many curves $D_{Q,p}(\mathbf{E}) > 1$.

The above questions also require bounds of various divisor sums.

3. Divisor sums

In fact, in this section we estimate sums which are slightly larger than those arising in our algorithm analysis, though the upper bounds are the same. As before we use $P(k)$ to denote the largest prime divisor of an integer $k \geqslant 1$ with the usual convention that $P(1) = 1$.

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

Let us consider the sum

$$R(n) = \sum_{d|n} \frac{P(d)^{1/2}}{d} .$$

LEMMA 3. *For all integers $n \geqslant 2$,*

$$R(n) \ll \big(\omega(n)\log\omega(6n)\big)^{1/2}$$

and hence

$$R(n) \ll (\log n)^{1/2} .$$

PROOF: We have

$$R(n) = \sum_{\ell|n} \ell^{1/2} \sum_{\substack{f|n/\ell \\ P(f)\leqslant \ell}} \frac{1}{f\ell} \leqslant \sum_{\ell|n} \ell^{1/2} \sum_{f|n/\ell} \frac{1}{f\ell} = \sum_{\ell|n} \ell^{-1/2} \sum_{f|n/\ell} \frac{1}{f}.$$

We have

$$\sum_{f|n/\ell} \frac{1}{f} \leqslant \sum_{f|n} \frac{1}{f} \leqslant \prod_{p|n} (1-1/p)^{-1} \leqslant \prod_{p\leqslant 6\omega(n)\log\omega(6n)} (1-1/p)^{-1} \ll \log\omega(6n) ,$$

where we have used a weak version of the Chebyshev bound for the $k-th$ prime together with the Chebyshev–Mertens formula for the product. Also, by Chebyshev's prime number bound and partial summation,

$$\sum_{\ell|n} \ell^{-1/2} \leqslant \sum_{\ell\leqslant 6\omega(n)\log\omega(6n)} \ell^{-1/2} \ll \omega(n)^{1/2}\big(\log\omega(6n)\big)^{-1/2},$$

giving the first statement of the lemma. The latter statement then follows immediately from (6). □

For real positive Q and α and an integer n, we define the sum

$$U_\alpha(Q,n) = \sum_{\ell|m\Rightarrow \ell|n,\, \ell\geqslant Q} \frac{1}{m^\alpha} \quad \text{and} \quad V_\alpha(Q,n) = \sum_{\substack{\omega(m)\geqslant 2 \\ \ell|m\Rightarrow \ell|n,\, \ell\geqslant Q}} \frac{1}{m^\alpha}.$$

LEMMA 4. *For any $\alpha > 0$, integer $n \geqslant 3$, and $Q \geqslant (\log n)^{1/\alpha}$, we uniformly have*

$$U_\alpha(Q,n) \ll Q^{-\alpha}\omega(n).$$

PROOF: We have,

$$U_\alpha(Q,n) = \prod_{\substack{\ell|n \\ \ell\geqslant Q}} \sum_{j=0}^{\infty} \ell^{-\alpha j} - 1 = \prod_{\substack{\ell|n \\ \ell\geqslant Q}} \left(1 + \frac{1}{\ell^\alpha - 1}\right) - 1$$

$$\leqslant \left(1 + \frac{1}{Q^\alpha - 1}\right)^{\omega(n)} - 1 \leqslant \exp\left(\frac{\omega(n)}{Q^\alpha - 1}\right) - 1.$$

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

By (6) we have $\omega(n) = o(Q^\alpha)$. Therefore

$$\exp\Big(\frac{\omega(n)}{Q^\alpha - 1}\Big) - 1 \ll \frac{\omega(n)}{Q^\alpha - 1},$$

which finishes the proof. □

LEMMA 5. *For any $\alpha > 0$, integer $n \geqslant 3$, and $Q \geqslant (\log n)^{1/\alpha}$, we uniformly have*

$$V_\alpha(Q, n) \ll Q^{-2\alpha}\omega(n)^2.$$

PROOF: Let

$$\sigma = \sum_{\substack{\ell \mid n \\ \ell \geqslant Q}} \frac{1}{\ell^\alpha - 1}$$

and note that by (6), under the condition $Q \geqslant (\log n)^{1/\alpha}$, we have

$$\sigma = O(\omega(n)Q^{-\alpha}) = o(1). \tag{7}$$

As in the proof of Lemma 4 we obtain

$$V_\alpha(Q, n) = \prod_{\substack{\ell \mid n \\ \ell \geqslant Q}} \Big(1 + \frac{1}{\ell^\alpha - 1}\Big) - 1 - \sigma.$$

Furthermore,

$$\log \prod_{\substack{\ell \mid n \\ \ell \geqslant Q}} \Big(1 + \frac{1}{\ell^\alpha - 1}\Big) = \sum_{\substack{\ell \mid n \\ \ell \geqslant Q}} \log\Big(1 + \frac{1}{\ell^\alpha - 1}\Big) \leqslant \sum_{\substack{\ell \mid n \\ \ell \geqslant Q}} \frac{1}{\ell^\alpha - 1} = \sigma.$$

Therefore,

$$V_\alpha(Q, n) \leqslant e^\sigma - 1 - \sigma \ll \sigma^2$$

by (7). Thus, we finish the proof. □

For a real positive K and Q, and an integer n, we define the sum

$$U_\alpha(K, Q, n) = \sum_{\substack{m \geqslant K \\ \ell \mid m \Rightarrow \ell \mid n, \ell \geqslant Q}} \frac{1}{m^\alpha}.$$

LEMMA 6. *For any integer $n \geqslant 2$, and reals $\alpha > 0$, $K > 1$, $Q > \big(\omega(n) + 1\big)^{1/\alpha}$, we uniformly have*

$$U_\alpha(K, Q, n) \ll K^{-\alpha + \log(\omega(n)+1)/\log Q}.$$

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

PROOF: Let

$$\beta = 1 - \alpha^{-1}\frac{\log(\omega(n)+1)}{\log Q},$$

so that $Q^{\alpha(1-\beta)} - 1 = \omega(n)$. As the hypothesis on Q implies that $\beta > 0$, we have

$$\begin{aligned} U_\alpha(K,Q,n) &\leqslant K^{-\alpha\beta} \sum_{\ell \mid m \Rightarrow \ell \mid n,\, \ell \geqslant Q} m^{-\alpha(1-\beta)} = K^{-\alpha\beta} \prod_{\substack{\ell \mid n \\ \ell \geqslant Q}} \sum_{j=0}^{\infty} \ell^{j\alpha(1-\beta)} \\ &= K^{-\alpha\beta} \prod_{\substack{\ell \mid n \\ \ell \geqslant Q}} \Bigl(1 + \frac{1}{\ell^{\alpha(1-\beta)} - 1}\Bigr) \leqslant K^{-\alpha\beta} \Bigl(1 + \frac{1}{Q^{\alpha(1-\beta)} - 1}\Bigr)^{\omega(n)} \\ &= K^{-\alpha\beta} \Bigl(1 + \frac{1}{\omega(n)}\Bigr)^{\omega(n)} \leqslant eK^{-\alpha\beta} = eK^{-\alpha + \log(\omega(n)+1)/\log Q}, \end{aligned}$$

which concludes the proof. □

4. AVERAGE COMPLEXITY OF FINDING THE GROUP STRUCTURE OF ELLIPTIC CURVES

We now estimate the average complexity of several "natural" combinations of smoothness tests and factorisation algorithms described in Section 2 for an optimally chosen value of Q.

THEOREM 7. *There is a deterministic algorithm which factors $D_p(\mathbf{E})$ whose average run time, taken over all over all elliptic curves $\mathbf{E}$ given by (1) with $a, b \in \mathbb{F}_p$, is*

$$T(p) \leqslant \omega(p-1)(\log p)^{2+o(1)}$$

PROOF: We simply start with using the Pollard-Strassen smoothness test as a factoring algorithm (and as a primality test too), see [17]. Thus from Lemma 2 (applied with $Q = 1$) we deduce

$$\begin{aligned} T(p) &\leqslant \log p (\log\log p)^2 \sum_{d \mid p-1} \min\bigl\{P(d)^{1/2}, d^{1/4}\bigr\} (\log d)^{1+o(1)} \bigl(d^{-1} + p^{-1/2}\bigr) \\ &\leqslant (\log p)^{2+o(1)} \sum_{d \mid p-1} \Bigl(\frac{P(d)^{1/2}}{d} + d^{1/4} p^{-1/2}\Bigr) \\ &\leqslant (\log p)^{2+o(1)} \bigl(R(p-1) + p^{-1/4}\tau(p-1)\bigr). \end{aligned}$$

Thus, by Lemma 3 and the bound (5), we conclude the proof. □

We now show that probabilistic algorithms lead to a better (in fact almost linear) bound on the average complexity of factoring $D_p(\mathbf{E})$.

THEOREM 8. *There is a probabilistic algorithm which factors $D_p(\mathbf{E})$ whose average expected run time, taken over all over all elliptic curves $\mathbf{E}$ given by (1) with $a, b \in \mathbb{F}_p$, is*

$$T(p) \leqslant (\log p)^{1+o(1)}.$$

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

PROOF: We set $Q = \exp\left((\log\log p)^{4/3}\right)$ and use the Lenstra–Pila–Pomerance hyperelliptic smoothness test [11], and then the Pollard-Strassen smoothness test as a factoring algorithm (and as a primality test too), see [17]. From Lemma 2 we derive

$$\begin{aligned} T(p) &\leqslant \exp\left(c_0(\log Q)^{2/3}(\log\log Q)^{1/3}\right)(\log p)^{1+o(1)} \\ &\quad + \log p(\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} d^{1/4+o(1)}\left(d^{-1} + p^{-1/2}\right) \\ &\leqslant \exp\left(c_0(\log Q)^{2/3}(\log\log Q)^{1/3}\right)(\log p)^{1+o(1)} \\ &\quad + (\log p)^{1+o(1)}\left(U_{2/3}(Q, p-1) + p^{-1/4+o(1)}\right). \end{aligned}$$

Thus by Lemma 4 and the bounds (5) and (6), we obtain

$$\begin{aligned} T(p) \leqslant \exp\left(c_0(\log Q)^{2/3}(\log\log Q)^{1/3}\right)(\log p)^{1+o(1)} \\ + (\log p)^{2+o(1)}\omega(p-1)Q^{-2/3} + o(1). \end{aligned}$$

For the above choice of Q we have

$$\exp\left(c_0(\log Q)^{2/3}(\log\log Q)^{1/3}\right) = (\log p)^{o(1)}$$

and

$$(\log p)^{2+o(1)}\omega(p-1)Q^{-2/3} = o(1)$$

which concludes the proof. □

5. Results for almost all curves

We now estimate the number of curves for which a smoothness test with a given threshold Q does not factor $D_p(\mathbf{E})$ completely.

THEOREM 9. *Given a smoothness bound Q, the proportion of elliptic curves $\mathbf{E}$ given by (1) with $a, b \in \mathbb{F}_p$ and such that $D_{Q,p}(\mathbf{E}) > 1$ is*

$$\rho(Q, p) \ll Q^{-1}\omega(p-1)(\log p)^{1+o(1)} + p^{-1/2+o(1)}.$$

PROOF: Using (2) and Lemma 1, as in the proof of Lemma 2, we obtain

$$\begin{aligned} \rho(Q, p) &\ll p^{-1/2}\log p(\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \sum_{\substack{|N-p-1| \leqslant 2p^{1/2} \\ d \mid N}} 1 \\ &\ll \log p(\log\log p)^2 \sum_{\substack{d \mid p-1 \\ \ell \mid d \Rightarrow \ell \geqslant Q}} d^{-1} + p^{-1/2}\tau(p-1)\log p(\log\log p)^2 \\ &\ll (\log p)^{1+o(1)}\left(U_1(Q, p-1) + p^{-1/2+o(1)}\right) \end{aligned}$$

Using Lemma 4 we finish the proof. □

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

THEOREM 10. *Given a smoothness bound Q, the proportion of elliptic curves $\mathbf{E}$ given by* (1) *with $a, b \in \mathbb{F}_p$ and such that $D_{Q,p}(\mathbf{E})$ has at least two distinct prime factors is*

$$\widetilde{\rho}(Q,p) \ll Q^{-2}\omega(p-1)^2(\log p)^{1+o(1)} + p^{-1/2+o(1)}.$$

PROOF: The proof is completely analogous to that of Theorem 9, except that we use Lemma 5 instead of Lemma 4. □

In particular, we see from Theorem 10 that using the Pollard–Strassen smoothness test [**17**] with $Q = (\log p)^{A+1}$ for some constant $A > 0$, and the deterministic primality test [**1, 13**], we actually find the complete factorisation of $D_p(\mathbf{E})$ in deterministic polynomial time for all but the proportion $(\log p)^{1-2A+o(1)}$ of elliptic curves over $\mathbb{F}_p$.

Furthermore, using the hyperelliptic smoothness test [**11**] with

$$Q = \exp\left((A+1)\frac{(\log\log p)^{3/2}}{(\log\log\log p)^2}\right)$$

we actually find a complete factorisation of $D_p(\mathbf{E})$ in expected polynomial time for all but the proportion $Q^{-2+o(1)}$ of elliptic curves over $\mathbb{F}_p$. Finally, with the heuristic elliptic curve test we have the same result with

$$Q = \exp\left((A+1)\frac{(\log\log p)^2}{\log\log\log p}\right).$$

We can actually get a result of similar quality using fully proved subroutines for finding smooth parts and general factoring.

THEOREM 11. *For any fixed $A > 0$, and sufficiently large p, there is a probabilistic algorithm to compute the exponent of an elliptic curve $\mathbf{E}$ over $\mathbb{F}_p$ such that the expected running time of the algorithm is polynomial except possibly for the proportion*

$$\vartheta(p) \leqslant \exp\left(-A\frac{(\log\log p)^2}{\log\log\log p}\right)$$

of elliptic curves $\mathbf{E}$ given by (1) *with $a, b \in \mathbb{F}_p$.*

PROOF: We put

$$Q = \exp\left((\log\log p)^{4/3}\right) \qquad \text{and} \qquad K = \exp\left((A+1)\frac{(\log\log p)^2}{\log\log\log p}\right)$$

We use the hyperelliptic smoothness test [**11**], which for the above value of Q runs in polynomial time. Now, provided $D_{Q,p}(\mathbf{E}) \leqslant K$ we use the subexponential probabilistic algorithm of [**12**] which for the above value of K factors $D_{Q,p}(\mathbf{E})$ in polynomial time.

Thus, using (2) and Lemma 1, as in the proof of Lemma 2, we obtain

$$\begin{aligned}\vartheta(p) &\ll p^{-1/2}\log p(\log\log p)^2 \sum_{\substack{d \mid p-1 \\ d \geqslant K \\ \ell \mid d \Rightarrow \ell \geqslant Q}} \left(\frac{4p^{1/2}}{d} + 1\right) \\ &\ll U_1(K, Q, p-1)\log p(\log\log p)^2 + p^{-1/2}\tau(p-1)\log p(\log\log p)^2.\end{aligned}$$

https://doi.org/10.1017/S0004972700035048 Published online by Cambridge University Press

Since $\log \omega(n) \ll \log\log n = (\log Q)^{3/4}$, Lemma 6 finishes the proof. □

Clearly, assuming that for some constant $C > 0$, one can factor an integer n within the expected number of $\exp\left(C(\log n)^{1/3}(\log\log n)^{2/3}\right)$ bit operations (for example, under the assumption that the number field sieve runs in the widely believed time), the bound of Theorem 11 can be improved to

$$\vartheta(p) = \exp\Big(-A\frac{(\log\log p)^3}{(\log\log\log p)^2}\Big).$$

6. Remarks

It immediately follows from the Bombieri-Vinogradov theorem, see [7], that for infinitely many primes p, $p-1$ has a divisor d with $p^{1/2} \geqslant d \geqslant p^{1/2+o(1)}$. Moreover, one has this with d the product of two primes of the same magnitude, and so might be expected to be hard to factor. By the classical result of Deuring [8], when $\mathbf{E}$ runs through the whole family (1) with $a, b \in \mathbb{F}_p$, the cardinality $\#\mathbf{E}(\mathbb{F}_p)$ takes all integer values, except for p, in the Hasse–Weil interval $[p+1-2p^{1/2}, p+1+2p^{1/2}]$, see also [3, 10, 20, 23]. Thus $D_p(\mathbf{E}) = p^{1/2+o(1)}$ for infinitely many pairs of primes p and curves $\mathbf{E}$. And in fact, $D_p(\mathbf{E})$ is infinitely often divisible by a number d which is $p^{1/2+o(1)}$ and is the product of two primes of the same magnitude. Moreover, it follows from [21, Proposition 3.3] that even if one has an arbitrary, but fixed curve $\mathbf{E}$ defined over the field of rational numbers, then, under the *Extended Riemann Hypothesis*, there are infinitely many primes p with $D_p(\mathbf{E}) \geqslant p^{1/8+o(1)}$ (to see this one has to recall that L in (3) satisfies $L \mid p-1$). We should remark, that in this example $D_p(\mathbf{E})$ is likely to be prime so its "factorisation" is an easy task. However, the method of [21] can easily be modified to prove that for infinitely many primes p, the value of $D_p(\mathbf{E})$ is a product of two large primes.

Thus it would seem that as long as the complexity of integer factorisation algorithms remain non-polynomial, the worst case complexity of the algorithm of [16] is not polynomial either.

It is easy to show using sieve methods that if $\psi(p) \to 0$ arbitrarily slowly as $p \to \infty$, then for almost all primes p, the number $p-1$ has two prime divisors q between $p^{\psi(p)}$ and $p^{1/4}$. Thus most primes p have an elliptic curve $\mathbf{E}$ over $\mathbb{F}_p$ with a value of $D_p(\mathbf{E})$ that is presumably very difficult to factor completely.

The same arguments also apply to elliptic curves over arbitrary finite fields. One only needs a generalisation of the bound from [10] to this case (which should be a rather straightforward task). We however, remark that for an ordinary curve $\mathbf{E}$ defined over a finite field of q elements, and considered in the consecutive extensions $\mathrm{E}(\mathbb{F}_{q^n})$ (which is a very typical scenario for cryptographic applications), the value of $D_{q^n}(\mathbf{E}) = \gcd(\mathbf{E}(\mathbb{F}_{q^n}), q^n - 1)$ is subexponential. Namely, as it follows from [15, Section 5], for any $\delta > 0$ there is an effectively computable absolute constant $C(\delta) > 0$ such that for

any $X > 1$

$$D_{q^n}(\mathbf{E}) \leqslant q^{C(\delta)n(\log n)^{-1/6}}$$

for all $n \leqslant X$ except at most $O(X^\delta)$ of them. On the other hand, it follows from [**14**] that

$$D_{q^n}(\mathbf{E}) \geqslant \exp\left(n^{c/\log\log n}\right)$$

for some absolute constant $c > 0$ and infinitely many n.

References

[1] M. Agrawal, N. Kayal and N. Saxena, 'PRIMES is in P', *Ann. of Math. (2)* **60** (2004), 781–793.

[2] R. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange, K. Nguyen and F. Vercauteren, *Elliptic and hyperelliptic curve crytography: Theory and practice* (CRC Press) (to appear).

[3] B.J. Birch, 'How the number of points of an elliptic curve over a fixed prime field varies', *J. Lond. Math. Soc.* **43** (1968), 57–60.

[4] I. Blake, G. Seroussi and N. Smart, *Elliptic curves in cryptography*, London Math. Soc. Lecture Note Series **265** (Cambridge Univ. Press, Cambridge, 1999).

[5] D. Coppersmith, 'Modifications to the number field sieve', *J. Cryptology* **6** (1993), 169–180.

[6] R. Crandall and C. Pomerance, *Prime numbers: A computational perspective* (Springer-Verlag, Berlin, 2001).

[7] H. Davenport, *Multiplicative number theory*, 2nd edition (Springer-Verlag, New York, 1980).

[8] M. Deuring, 'Die Typen der Multiplikatorenringe elliptischer Funktionenkörper', *Abh. Math. Sem. Hansischen Univ.* **14** (1941), 197–272.

[9] D.R. Kohel and I.E. Shparlinski, *Exponential sums and group generators for elliptic curves over finite fields*, Lect. Notes in Comp. Sci. **1838** (Springer-Verlag, Berlin, 2000), **pp.** 395–404.

[10] H.W. Lenstra, Jr., 'Factoring integers with elliptic curves', *Annals of Math.* **126** (1987), 649–673.

[11] H.W. Lenstra, Jr., J. Pila and C. Pomerance, 'A hyperelliptic smoothness test, I', *Philos. Trans. Royal Soc. London, Ser. A.* **345** (1993), 397–408.

[12] H.W. Lenstra, Jr. and C. Pomerance, 'A rigorous time bound for factoring integers', *J. Amer. Math. Soc.* **5** (1992), 483–516.

[13] H.W. Lenstra, Jr. and C. Pomerance, 'Primality testing with Gaussian periods', (in preparation).

[14] F. Luca, J. McKee and I.E. Shparlinski, 'Small exponent point groups on elliptic curves', *J. Théor. Nombres Bordeaux* (to appear).

[15] F. Luca and I.E. Shparlinski, 'On the exponent of the group of points on elliptic curves in extension fields', *Internat. Math. Res, Notices* (to appear).

[16] V.S. Miller, 'The Weil pairing, and its efficient calculation', *J. Cryptology* **17** (2004), 235–261.

[17] C. Pomerance, 'Analysis and comparison of some integer factoring algorithms', in *Computational Methods in Number Theory, Part I*, (H.W. Lenstra, Jr. and R. Tijdeman, Editors), Math. Centre Tracts **154** (Math Centrum, Amsterdam, 1982), **pp.** 89–139.

[18] K. Prachar, *Primzahlverteilung* (Springer-Verlag, Berlin, 1957).

[19] R. Schoof, 'Elliptic curves over finite fields and the computation of square roots mod p', *Math. Comp.* **44** (1985), 483–494.

[20] R. Schoof, 'Nonsingular plane cubic curves over finite fields', *J. Combin. Theory, Ser.A* **47** (1987), 183–211.

[21] R. Schoof, 'The exponents of the group of points on the reduction of an elliptic curve', in *Arithmetic Algebraic Geometry*, Progr. Math. **89** (Birkhäuser, Boston, MA, 1991), **pp.** 325–335.

[22] J.H. Silverman, *The arithmetic of elliptic curves* (Springer-Verlag, Berlin, 1995).

[23] W.C. Waterhouse, 'Abelian varieties over finite fields', *Ann. Sci. Ecole Norm. Sup.* **2** (1969), 521–560.

Department of Mathematics
University of Toronto
Toronto, Ontario M5S 3G3
Canada
e-mail: frdlndr@math.toronto.edu

Department of Mathematics
Dartmouth College
Hanover, NH 03755-355
United States of America
e-mail: carlp@gauss.dartmouth.edu

Department of Computing
Macquarie University
Sydney, NSW 2109
Australia
e-mail: igor@ics.mq.edu.au