8.1 Point counting

We now consider the problem of determining the number of points on an elliptic curve E over a finite field $\mathbb{F}_q$. The most naïve approach one could take would be to evaluate the curve equation $y^2 = x^3 + Ax + B$ for E at every pair $(x, y) \in \mathbb{F}_q^2$, count the number of solutions, and add 1 for the point at infinity. This takes $O(q^2\mathsf{M}(\log q))$ time. Note that the input to this problem is the pair of coefficients $A, B \in \mathbb{F}_q$, which each have $O(n)$ bits, where $n = \log q$. Thus in terms of the size of its input, this algorithm takes

$$O\big(\exp(2n)\mathsf{M}(n)\big)$$

time, which is exponential in n.

A slightly less naïve approach is to precompute a table of quadratic residues in $\mathbb{F}_q$ so that we can very quickly compute the extended Legendre symbol $\left(\frac{\cdot}{\mathbb{F}_q}\right)$ defined in the last lecture. We can construct such a table in $O(q\mathsf{M}(\log q))$ time, and we can then compute

$$\#E(\mathbb{F}_q) = q + 1 + \sum_{x \in \mathbb{F}_q} \left(\frac{x^3 + Ax + B}{\mathbb{F}_q}\right)$$

in $O(q\mathsf{M}(\log q))$ time, yielding a total running time of

$$O(\exp(n)\mathsf{M}(n)).$$

But we still have not taken advantage of Hasse's theorem, which tells us that the quantity $\#E(\mathbb{F}_q)$ that we wish to compute must lie in the *Hasse interval*

$$\mathcal{H}(q) = \big[q + 1 - 2\sqrt{q},\ q + 1 + 2\sqrt{q}\big] = [(\sqrt{q} - 1)^2, (\sqrt{q} + 1)^2],$$

which has width $4\sqrt{q}$.

8.2 Computing the order of a point

Before giving an algorithm to compute $\#E(\mathbb{F}_q)$ using Hasse's theorem, let us first consider an easier problem: computing the order of a single point $P \in E(\mathbb{F}_q)$. Since the order of the group $E(\mathbb{F}_q)$ lies in $\mathcal{H}(q)$, we know that $\mathcal{H}(q)$ contains at least one integer M such that $MP = 0$, and any such M is a multiple of $|P|$. To find such an M, we set $M_0 = \lceil(\sqrt{q}-1)^2\rceil$, compute M_0P, and then generate the sequence of points

$$M_0P, (M_0 + 1)P, (M_0 + 2)P, \ldots, MP = 0,$$

using repeated addition by P.[1] Note that M is bounded by $M_0 + 4\sqrt{q}$.

We then compute the prime factorization $M = p_1^{e_1} \cdots p_w^{e_w}$, which is easy compared to the time to find M, and compute the exact order of the point P using the following generic algorithm.

[1] Of course the initial scalar multiplication M_0P should be performed using one of the generic exponentiation algorithms we saw in Lecture 4, not by adding P to itself M_0 times!

Algorithm 8.1. Given an element P of an additive group and the prime factorization $M = p_1^{e_1} \cdots p_k^{e_k}$ of an integer M for which $MP = 0$, compute the order of P as follows:

1. Let $m = M = p_1^{e_1} \cdots p_k^{e_k}$.
2. For each prime p_i, while $p_i | m$ and $(m/p_i)P = 0$, replace m by m/p_i.
3. Output m.

When this procedure is complete we know that $mP = 0$ and $(m/p) \neq 0$ for every prime p dividing m; this implies that $m = |P|$. You will analyze the efficiency of this algorithm and develop several improvements to it in Problem Set 4, but the number of group operations is clearly polynomial in $\log M$.

Thus the time to compute $|P|$ is dominated by the time to find a multiple of $|P|$ in $\mathcal{H}(q)$. This involves $O(\sqrt{q})$ operations in $E(\mathbb{F}_q)$, yielding a bit complexity of $O(\sqrt{q}\ \mathsf{M}(\log q))$ or

$$O\big(\exp(n/2)\mathsf{M}(n)\big).$$

We will shortly see how this can be further improved, but first we consider how we may use our algorithm for computing $|P|$ to compute $\#E(\mathbb{F}_q)$. If we are lucky (and when q is large we usually will be), the multiple M of $|P|$ that we find will actually be the *unique* multiple of $|P|$ in $\mathcal{H}(q)$. If this happens, then we must have $M = \#E(\mathbb{F}_q)$. Otherwise, we might try our luck with a different point P. If we can find any combination of points such that the least common multiple of their orders has a unique multiple in $\mathcal{H}(q)$, then we can determine the group order.

Now this won't always be possible. But before addressing that issue, let us consider the question of how long it might take to compute the least common multiple of the orders of *all* the points in $E(\mathbb{F}_q)$, which is less than one might think.

8.3 The group exponent

Definition 8.2. For a finite group G, the *exponent* of G, denoted $\lambda(G)$, is defined by

$$\lambda(G) = \operatorname{lcm}\{|\alpha| : \alpha \in G\}.$$

Note that $\lambda(G)$ is a divisor of $|G|$ and is divisible by the order of every element of G. Thus $\lambda(G)$ is the maximal possible order of an element of G, and when G is abelian this maximum is achieved: there exists an element with order $\lambda(G)$. To see this, note that the structure theorem for abelian groups allows us to decompose G as a sum of cyclic groups

$$G \simeq \mathbb{Z}/n_1\mathbb{Z} \oplus \mathbb{Z}/n_2\mathbb{Z} \oplus \cdots \oplus \mathbb{Z}/n_r\mathbb{Z},$$

with $n_i | n_{i+1}$ for $1 \leq i < r$. Thus $\lambda(G) = n_r$, and any generator for $\mathbb{Z}/n_r\mathbb{Z}$ has order $\lambda(G)$.

If we compute the least common multiple of a sufficiently large subset of a finite abelian group G we will eventually obtain $\lambda(G)$. If we pick points at random, how many points do we expect to need in order to obtain $\lambda(G)$? The answer is surprisingly small: two points are usually enough.

Theorem 8.3. *Let G be a finite abelian group with exponent $\lambda(G)$. Let α and β be uniformly distributed random elements of G. Then*

$$\Pr[\operatorname{lcm}(|\alpha|, |\beta|) = \lambda(G)] > \frac{6}{\pi^2}.$$

Proof. We first reduce to the case that G is cyclic. As noted above, G is isomorphic to a direct sum of cyclic groups $C_1 \oplus C_2 \oplus \cdots \oplus C_r$, where C_r has order $\lambda(G)$. Let α_r and β_r be the projection of α and β to C_r. Then $\operatorname{lcm}(|\alpha_r|, |\beta_r|) = \lambda(G)$ implies $\operatorname{lcm}(|\alpha|, |\beta|) = \lambda(G)$, and therefore

$$\Pr[\operatorname{lcm}(|\alpha|, |\beta|) = \lambda(G)] \geq \Pr[\operatorname{lcm}(|\alpha_r|, |\beta_r|) = \lambda(G)].$$

So we now assume that G is cyclic with generator γ. Let $p_1^{e_1} \cdots p_k^{e_k}$ be the prime factorization of $\lambda(G)$. Let $\alpha = a\gamma$, with $0 \leq a < \lambda(G)$. Unless a is a multiple of p_i, which occurs with probability $1/p_i$, the order of α will be divisible by $p_i^{e_i}$, and similarly for β. These two probabilities are independent, thus with probability $1 - 1/p_i^2$ at least one of α and β has order divisible by $p_i^{e_i}$. Call this event E_i. The events $E_1, \ldots, E_k$ are independent, since we may write G as a direct sum of cyclic groups of prime-power orders $p_1^{e_1}, \ldots, p_k^{e_k}$, and the projections of α and β to each of these cyclic groups are uniformly and independently distributed. Thus

$$\begin{aligned}\Pr[\operatorname{lcm}(|\alpha|, |\beta|) = \lambda(G)] &= \Pr[E_1 \cap \cdots \cap E_k] \\ &= \prod_{p|\lambda(G)} (1 - p^{-2}) > \prod_p (1 - p^{-2}) = \left(\sum_{n=1}^{\infty} \frac{1}{n^2}\right)^{-1} = \frac{1}{\zeta(2)} = \frac{6}{\pi^2},\end{aligned}$$

where $\zeta(s) = \sum n^{-s}$ is the Riemann zeta function. □

Theorem 8.3 implies that if we generate random points $P \in E(\mathbb{F}_q)$ and accumulate the least common multiple N of their orders, we should expect to obtain $\lambda(E(\mathbb{F}_q))$ within $O(1)$ iterations. Regardless of when we obtain $\lambda(E(\mathbb{F}_q))$, at every stage we know that N divides $\#E(\mathbb{F}_q)$, and if we ever find that N has a unique multiple M in the Hasse interval $\mathcal{H}(q)$, then we know that $\#E(\mathbb{F}_q) = M$. Unfortunately this might not ever happen; it could be that $\lambda(E(\mathbb{F}_q))$ is smaller than $4\sqrt{q}$, in which case it might have more than one multiple in $\mathcal{H}(q)$. To deal with this problem we need to consider the *quadratic twist* of E, which you saw in Problem Set 1.

8.4 The quadratic twist of an elliptic curve

Suppose d is an element of $\mathbb{F}_q$ that is *not* a quadratic residue, so that $\left(\frac{d}{\mathbb{F}_q}\right) = -1$. If we consider the elliptic curve $\tilde{E}$ defined by $dy^2 = x^3 + Ax + B$, then there will be a point of the form (x, y) on the curve if and only if $x^3 + Ax + B$ is *not* a quadratic residue. Thus

$$\#\tilde{E}(\mathbb{F}_q) = q + 1 - \sum_{x \in \mathbb{F}_q} \left(\frac{x^3 + Ax + B}{\mathbb{F}_q}\right),$$

and it follows that if $\#E(\mathbb{F}_q) = q+1-t$, then $\#\tilde{E}(\mathbb{F}_q) = q+1+t$. The curve $\tilde{E}$ is called the *quadratic twist* of E (by d). We can put the curve equation for $\tilde{E}$ in standard Weierstrass form by substituting x/d for x and y/d^2 for y and then clearing denominators, yielding

$$y^2 = x^3 + d^2Ax + d^3B.$$

Notice that it does not matter which non-residue d we choose. As you showed in Problem Set 1, if d and d' are any two non-residues in $\mathbb{F}_q$, then the corresponding curves $\tilde{E}$ and $\tilde{E}'$ are isomorphic over $\mathbb{F}_q$, thus we refer to $\tilde{E}$ as "the" quadratic twist of E.[2]

[2]This situation is specific to finite fields. Over $\mathbb{Q}$ for example, every elliptic curve has infinitely many quadratic twists that are not isomorphic over $\mathbb{Q}$ (of course they are all isomorphic over $\overline{\mathbb{Q}}$).

Our interest in the quadratic twist of E lies in the fact that

$$\#E(\mathbb{F}_q) + \#\tilde{E}(\mathbb{F}_q) = 2q + 2.$$

Thus if we can compute either $\#E(\mathbb{F}_q)$ or $\#\tilde{E}(\mathbb{F}_q)$, we can easily determine both values.

8.5 Mestre's Theorem

As noted above, it is not necessarily the case that the exponent of $E(\mathbb{F}_p)$ has a unique multiple in the Hasse interval. But if we also consider the quadratic twist $\tilde{E}(\mathbb{F}_p)$, then a theorem of Mestre (published by Schoof in [4]) ensures that for all primes $p > 229$, either $\lambda(E(\mathbb{F}_p))$ or $\lambda(\tilde{E}(\mathbb{F}_p))$ has a unique multiple in the Hasse interval $\mathcal{H}(p)$. There is a generalization of this theorem that works for arbitrary prime powers q, see [2], but we will restrict ourselves to the case where $q = p$ is prime.

Theorem 8.4 (Mestre)**.** *Let $p > 229$ be prime, and let $E/\mathbb{F}_p$ be an elliptic curve with quadratic twist $\tilde{E}/\mathbb{F}_p$. Then either $\lambda(E(\mathbb{F}_p))$ or $\lambda(\tilde{E}(\mathbb{F}_p)$ has a unique multiple in $\mathcal{H}(p)$.*

Proof. Let $E(\mathbb{F}_p) \simeq \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/N\mathbb{Z}$ and $\tilde{E}(\mathbb{F}_p) \simeq \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/M\mathbb{Z}$, where $n|N$ and $m|M$. Let t be the trace of the Frobenius endormophism π of E. We have $E[n] = E(\mathbb{F}_p)[n]$, so π fixes $E[n]$ and the matrix π_n is the identity. The matrix π_{n^2} then has the form

$$\pi_{n^2} = \begin{bmatrix} 1 + an & bn \\ cn & 1 + dn \end{bmatrix},$$

for some $a, b, c, d \in \mathbb{Z}/n^2\mathbb{Z}$. We then have

$$p \equiv \det \pi_{n^2} \equiv 1 + (a + d)n \bmod n^2,$$
$$t \equiv \operatorname{tr} \pi_{n^2} \equiv 2 + (a + d)n \bmod n^2.$$

Thus $4p - t^2 \equiv 0 \bmod n^2$. The trace of Frobenius for $\tilde{E}$ is $-t$, and we similarly obtain $4p - t^2 \equiv 0 \bmod m^2$. Thus $\gcd(m^2, n^2)$ divides $4p - t^2$. We also have $t = un + 2$ and $t = vm - 2$, for some integers u and v, and subtracting these equations yields $un - vm = 4$. This implies $\gcd(m, n) \leq 4$, and therefore $\gcd(m^2, n^2) \leq 16$. Thus

$$\frac{m^2 n^2}{16} \leq 4p - t^2 \leq 4p. \tag{1}$$

Suppose for the sake of contradiction that $N = \lambda(E(\mathbb{F}_p))$ and $M = \lambda(\tilde{E}(\mathbb{F}_p))$ both have more than one multiple in $\mathcal{H}(p)$. Then M and N are both at most $4\sqrt{p}$, so $MN \leq 16p$. Since mM and nN lie in $\mathcal{H}(p)$, they are both greater than $(\sqrt{p}-1)^2$, hence $mnMN \geq (\sqrt{p}-1)^4$. It follows that $mn \geq (\sqrt{p}-1)^4/(16p)$. Dividing by 4 and squaring both sides yields

$$\frac{m^2 n^2}{16} \geq \frac{(\sqrt{p}-1)^8}{4096p^2}. \tag{2}$$

Combining (1) and (2), we have

$$16384p^3 \geq (\sqrt{p}-1)^8. \tag{3}$$

This implies that if neither M nor N has a unique multiple in $\mathcal{H}(p)$, then $p < 17413$. An exhaustive computer search for $p < 17413$ finds that in fact we must have $p \leq 229$ □

8.6 Computing the group order with Mestre's Theorem

We now give a complete algorithm to compute $\#E(\mathbb{F}_p)$ using Mestre's theorem, assuming that p is a prime greater than 229 (if p is smaller than this we can easily count points using one of our naïve algorithms); see [2] for an analogous algorithm that works for all prime powers $q > 49$. As usual, $\mathcal{H}(p) = [(\sqrt{p}-1)^2, (\sqrt{p}+1)^2]$ denotes the Hasse interval.

Algorithm 8.5. Given $E/\mathbb{F}_p$ with $p > 229$ prime, compute $\#E(\mathbb{F}_p)$ as follows:

1. Compute a quadratic twist $\tilde{E}$ of E using a randomly chosen non-residue $d \in \mathbb{F}_p$.
2. Let $E_0 = E$ and $E_1 = \tilde{E}$, let $N_0 = N_1 = 1$, and let $i = 0$.
3. While neither N_0 nor N_1 has a unique multiple in $\mathcal{H}(p)$:
   a. Generate a random point $P \in E_i(\mathbb{F}_p)$.
   b. Find an integer $M \in \mathcal{H}(p)$ such that $MP = 0$.
   c. Factor M and compute $|P|$ via Algorithm 8.1.
   d. Replace N_i by $\operatorname{lcm}(N_i, |P|)$ and replace i by $1 - i$.
4. If N_0 has a unique multiple M in $\mathcal{H}(p)$ then return M, otherwise return $2p + 2 - M$, where M is the unique multiple of N_1 in $\mathcal{H}(p)$.

It is clear that the output of the algorithm is correct, and it follows from Theorems 8.3 and 8.4 that the expected number of iterations of step 3 is $O(1)$. Thus we have a *Las Vegas* algorithm to compute $\#E(\mathbb{F}_p)$. Its running time is dominated by the time to find M in step 3b, and we obtain a total expected running time of $O(\sqrt{p}\,\mathsf{M}(\log p))$, or

$$O\big(\exp(n/2)\mathsf{M}(n)\big).$$

We now show how this complexity can be further improved using the *baby-steps giant-steps* method to compute M in step 3b.

8.7 Baby-steps giant-steps

The baby-steps giant-steps method is a generic group algorithm first introduced by Dan Shanks in [5] and subsequently generalized by many authors. In the context of searching $\mathcal{H}(q) = [a, b]$ for an integer M such that $MP = 0$, it works as follows.

Algorithm 8.6. Given $P \in E(\mathbb{F}_q)$ compute $M \in \mathcal{H}(q)$ such that $MP = 0$:

1. Pick integers r and s such that $rs \geq b - a$.
2. Compute the set $S_{\text{baby}} = \{0, P, 2P, \ldots, (r-1)P\}$ of *baby steps*.
3. Compute the set $S_{\text{giant}} = \{aP, (a+r)P, (a+2r)P, \ldots, (a+(s-1)r)P\}$ of *giant steps*.
4. For each giant step $P_{\text{giant}} = (a + ir)P \in S_{\text{giant}}$, check whether $P_{\text{giant}} + P_{\text{baby}} = 0$ for some baby step $P_{\text{baby}} = jP \in S_{\text{baby}}$. If so, output $M = a + ri + j$.

When the algorithm terminates, we necessarily have $MP = 0$, so M is a multiple of $|P|$. Moreover, since *every* integer in $\mathcal{H}(q)$ can be written in the form $a + i + jr$ with $0 \le i < r$ and $0 \le j < s$, the algorithm is guaranteed to find such an M.

To implement this algorithm efficiently, we typically store the baby steps S_{baby} in a lookup table (such as a hash table or binary tree) and as each giant step P_{giant} is computed, we lookup $-P_{\text{giant}}$ in this table. Alternatively, one may compute the sets S_{baby} and S_{giant} in their entirety, sort both sets, and then efficiently search for a match. In both cases, we assume that the points in S_{baby} and S_{giant} are uniquely represented. If we are using projective coordinates (which makes the group operations more efficient), we must then convert each point to affine form; the point $(x : y : z)$ is put in the form $(x/z : y/z : 1)$ by computing the inverse of z in $\mathbb{F}_q$. Done naïvely, this requires $r + s$ field inversions, which costs $O((r + s)\mathsf{M}(n) \log n)$, but by using the method described in the next section, it is possible to perform $r + s$ field inversions in $O((r + s)\mathsf{M}(n))$ time. Assuming this is done, if we choose $r \approx s \approx 2q^{1/4}$, then the running time of the algorithm above is $O(q^{1/4}\mathsf{M}(\log q))$. If we use the baby-steps giant-steps method to implement step 3b of Algorithm 8.5, we can compute $\#E(\mathbb{F}_q)$ in

$$O\big(\exp(n/4)\mathsf{M}(n)\big)$$

expected time.

8.8 Batching field inversions

Suppose we are given a list of elements $\alpha_1, \ldots, \alpha_m \in \mathbb{F}_q$ whose inverses we wish to compute. The following algorithm accomplishes this using just one field inversion.

Algorithm 8.2 Given $\alpha_1, \ldots, \alpha_m \in \mathbb{F}_q$ compute $\alpha_1^{-1}, \ldots, \alpha_m^{-1}$ as follows:

1. Set $\beta_0 = 1$ and $\beta_i = \beta_{i-1}\alpha_i$ for i from 1 to m.
2. Compute $\gamma = \beta_n^{-1}$.
3. For i from n down to 1:
   a. Compute $\alpha_i^{-1} = \beta_{i-1}\gamma$.
   b. Replace γ by $\gamma\alpha_i$.

Note that at step 3a we have $\beta_i = \alpha_1 \cdots \alpha_i$ and $\gamma = \alpha_1^{-1} \cdots \alpha_i^{-1}$, which implies that $\alpha_i^{-1} = \beta_{i-1}\gamma$ as desired. The algorithm performs less than $3m$ multiplications in $\mathbb{F}_q$ and just one inversion in $\mathbb{F}_q$. Provided that $m = \Omega(\log n)$, its running time is $O(m\mathsf{M}(n))$.

In the context of Algorithm 8.1, if we are using a table of baby steps, we can compute all of the baby steps using projective coordinates, convert them to affine form using just one field inversion, and then construct the lookup table. For the giant steps we work in batches of size $m > \log n$, converting an entire batch to affine form using one field inversion and then performing table lookups.

8.9 Optimizations

There are a wide range of optimizations to the baby-steps giant-steps method that have been developed over the years. Here we mention just a few.

1. **Optimize expected time**: If we suppose that M is uniformly distributed over an interval of width N, then we should use $r \approx \sqrt{N/2}$ baby steps so that the average number of giant steps is $s \approx \sqrt{2N}/2 = \sqrt{N/2}$.

2. **Optimize for known distribution**: In the case of elliptic curves we know that M is *not* uniformly distributed, it has a semi-circular distribution.[3] This means we should search from the middle outwards by taking our first giant step in the middle of the interval (at $q+1$), and then alternating steps on either side. We should also choose the number of baby steps to optimize the expected time, using the fact that the expected distance between M and the middle of the interval is $\frac{8}{3\pi}\sqrt{q}$.

3. **Fast inverses**: In groups such as $E(\mathbb{F}_q)$ where we can compute inverses very quickly (the inverse of the point (x, y) is just $(x, -y)$), it makes sense to compute $-P_{\text{giant}}$ at the same time we compute P_{giant} and see whether either matches a baby step, equivalently, whether $P_{\text{giant}} \pm P_{\text{baby}} = 0$ holds. The number of baby steps should then be decreased by a factor of $\sqrt{2}$ to keep the time for baby steps and giant steps roughly equal.

4. **Parity**: We can easily determine the parity of $\#E(\mathbb{F}_q)$ by checking whether it has a point of order 2. If the curve equation is $y^2 = f(x) = x^3 + Ax + B$, then $\#E(\mathbb{F}_q)$ has even parity if and only if $f(x)$ has a root in $\mathbb{F}_q$ (recall that points of order 2 have y-coordinate 0), which we can determine using a root-finding algorithm.[4] Once we know the parity of M we can modify Algorithm 8.1 to only use baby steps that correspond to multiples of P with the same parity (so if M is odd we compute baby steps $P, 3P, 5P, \ldots$, adding $2P$ to each previous step), and use giant steps with even parity. We should then reduce the number of baby steps by a factor of $\sqrt{2}$.

References

[1] W. Banks and I. Shparlinski, *Sato-Tate, cyclicity, and divisibility statistics on average for elliptic curves of small height*, Israel J. Math. **173** (2009), pp. 253–277.

[2] J. E. Cremona and A.V. Sutherland, *On a theorem of Mestre and Schoof*, Journal de Théorie des Nombres de Bordeaux **22** (2010), pp. 353–358.

[3] J. von zur Gathen and J. Garhard, *Modern Computer Algebra*, second edition, Cambridge University Press, 2003.

[4] R. Schoof, *Counting points on elliptic curves over finite fields*, Journal de Théorie des Nombres de Bordeaux **7** (1995), pp. 219–254.

[5] D. Shanks, *Class number, a theory of factorization and genera*, in Proc. Symp. Pure Math., AMS, 1971, pp. 415–440.

[6] J. H. Silverman, *The Arithmetic of Elliptic Curves*, Graduate Texts in Mathematics **106**, second edition, Springer 2009.

[7] L. C. Washington, *Elliptic Curves: Number Theory and Cryptography*, second edition, Chapman and Hall/CRC, 2008.

[3] This follows from results showing that the Sato-Tate conjecture holds "on average"; see [1].

[4] In fact we only need to check whether $\gcd(x^q - x, f(x))$ has positive degree.