18.783 Elliptic Curves
Lecture 7

Andrew Sutherland

March 10, 2021
Hasse’s theorem

**Definition (from Lecture 6)**

If $\alpha$ is an isogeny, the **dual isogeny** $\hat{\alpha}$ is the unique isogeny for which $\hat{\alpha} \circ \alpha = [\deg \alpha]$. The **trace** of $\alpha \in \text{End}(E)$ is $\text{tr} \alpha := \alpha + \hat{\alpha} = 1 + \deg \alpha - \deg(1 - \alpha) \in \mathbb{Z}$.

**Theorem (Hasse, 1933)**

Let $E/\mathbb{F}_q$ be an elliptic curve over a field over a finite field. Then

$$\#E(\mathbb{F}_q) = q + 1 - \text{tr} \pi_E,$$

where the trace of the Frobenius endomorphism $\pi_E$ satisfies $|\text{tr} \pi_E| \leq 2\sqrt{q}$.

**Definition**

The **Hasse interval** $\mathcal{H}(q)$ is $[q + 1 - 2\sqrt{q}, q + 1 + 2\sqrt{q}] = [(\sqrt{q} - 1)^2, (\sqrt{q} + 1)^2]$
The Legendre symbol

Definition

For odd primes $p$ the Legendre symbol is defined by

$$\left( \frac{a}{p} \right) = \begin{cases} 
1 & \text{if } y^2 = a \text{ has two solutions mod } p \\
0 & \text{if } y^2 = a \text{ has one solution mod } p \\
-1 & \text{if } y^2 = a \text{ has no solutions mod } p
\end{cases} = \#\{\alpha \in \mathbb{F}_p : \alpha^2 = a\} - 1.$$

We also define $\left( \frac{a}{\mathbb{F}_q} \right)$ for $a \in \mathbb{F}_q$ with $q$ odd; just replace $\mathbb{F}_p$ with $\mathbb{F}_q$.

For $E : y^2 = x^3 + Ax + B$ over $\mathbb{F}_q$ we have

$$\#E(\mathbb{F}_q) = 1 + \sum_{x \in \mathbb{F}_q} 1 + \frac{x^3 + Ax + B}{\mathbb{F}_q} = q + 1 + \sum_{x \in \mathbb{F}_q} \frac{x^3 + Ax + B}{\mathbb{F}_q}.$$
Naive point counting

Let $E: y^2 = x^3 + Ax + B$ be an elliptic curve over $\mathbb{F}_q$. Computing $\#E(\mathbb{F}_q)$ via

$$\#E(\mathbb{F}_q) = 1 + \# \left\{ (x, y) \in \mathbb{F}_q^2 : y^2 = x^3 + Ax + B \right\}$$

take $O(q^2 M(\log q))$ time, which in terms of $n = \log q$ is $O(\exp(2n)M(n))$. But

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

can be computed in $O(\exp(n)M(n))$ time by precomputing a table of squares in $\mathbb{F}_q$.

But $\#E(\mathbb{F}_p)$ lies in the Hasse interval $\mathcal{H}(q)$ of width $4\sqrt{q}$. Surely we can do better!
Computing the order of a point

The order $|P|$ of any $P \in E(\mathbb{F}_q)$ divides $\#E(\mathbb{F}_q) \in \mathcal{H}(q) = [(\sqrt{q} - 1)^2, (\sqrt{q} + 1)^2]$. If we put $M_0 = [(\sqrt{q} - 1)^2]$, we can find a multiple $M$ of $|P|$ in $\mathcal{H}(q)$ by computing

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

We have $M \leq M_0 + 4\sqrt{q}$, so this takes $O(\sqrt{q} \log q) = O(\exp(n/2)M(n))$ time.

**Algorithm (Fast order computation)**

Given $P \in E(\mathbb{F}_q)$ and $M \in \mathcal{H}(q)$ such that $MP = 0$, compute $|P|$ as follows:

1. Compute $M = p_1^{e_1} \cdots p_r^{e_r}$ and set $m := M$.
2. For each prime $p_i$, while $p_i | m$ and $(m/p_i)P = 0$, replace $m$ by $m/p_i$.
3. Output $|P| = m$.

This algorithm takes much less than $O(\exp(n/2)M(n))$ time.

(in fact $O(\exp(n/5)n^{16/5})$ deterministically and $\exp(n^{1/2+o(1)})$ probabilistically).
The exponent of a group

**Definition**

The **exponent** of a finite group $G$ is $\lambda(G) := \text{lcm}\{|g| : g \in G\}$.

**Lemma**

*Let $G$ be a finite abelian group. Then $\exists g \in G$ such that $|g| = \lambda(G)$.***

**Proof:** Put $G \simeq \mathbb{Z}/n_1\mathbb{Z} \oplus \cdots \oplus \mathbb{Z}/n_r\mathbb{Z}$ with $n_i | n_{i+1}$ and take any generator of $\mathbb{Z}/n_r\mathbb{Z}$.

**Theorem**

*Let $G$ be a finite abelian group. If $g$ and $h$ are uniformly distributed elements of $G$ then

$$\Pr[\text{lcm}(|g|, |h|) = \lambda(G)] > \frac{6}{\pi^2}.$$*

**Proof:**

$$\Pr[\text{lcm}(|g|, |h|) = \lambda(G)] \geq \prod_{p | \lambda(G)} (1 - p^{-2}) \geq \prod_{p} (1 - p^{-2}) = \zeta(2)^{-1} = \frac{6}{\pi^2}.$$
Counting points on quadratic twists

Let $E : y^2 = x^3 + Ax + B$ be an elliptic curve over $\mathbb{F}_q$ and pick $s \in \mathbb{F}_q$ so $\left( s_{\mathbb{F}_q} \right) = -1$.

Then $\tilde{E} : sy^2 = x^3 + Ax + B$ is a (non-isomorphic) quadratic twist of $E$, and we have

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

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

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

To compute $#E(\mathbb{F}_q)$ it suffices to compute either $#E(\mathbb{F}_q)$ or $#\tilde{E}(\mathbb{F}_q)$.

We can put $\tilde{E}$ in Weierstrass form as $\tilde{E} : y^2 = x^3 + s^2 Ax + s^3 B$. 

Mestre’s theorem/algorithm

**Theorem (Mestre)**

Let $p > 229$ be prime, $E / \mathbb{F}_p$ an elliptic curve with quadratic twist $\tilde{E} / \mathbb{F}_p$. At least one of $\lambda(E(\mathbb{F}_p))$ and $\lambda(\tilde{E}(\mathbb{F}_p))$ has a unique multiple in $\mathcal{H}(p)$.

**Algorithm (Mestre)**

Given $E / \mathbb{F}_p$ with $p > 229$ compute $E(\mathbb{F}_p)$ as follows:

1. Compute $\tilde{E}$, and set $E_0 := E$, $E_1 := \tilde{E}$, $N_0 := 1$, $N_1 := 1$, $i := 0$.
2. While neither $N_0, N_1$ has a unique multiple $U_0, U_1$ in $\mathcal{H}(p)$:
   - a. Pick a random $P \in E_i(\mathbb{F}_p)$ and compute $M \in \mathcal{H}(p)$ such that $MP = 0$.
   - b. Use $M$ to compute $|P|$, then replace $N_i$ with $\text{lcm}(N_i, |P|)$ and replace $i$ by $1 - i$.
3. Output $\#E(\mathbb{F}_p) = U_0$ or $\#E(\mathbb{F}_p) = 2p + 2 - U_1$ (whichever is defined).

We expect $O(1)$ iterations in Step 2, expected running time is $O(\exp(n/2)M(n))$.  

8
Algorithm (Shanks)

Given \( P \in E(\mathbb{F}_q) \) compute \( M \in \mathcal{H}(q) \) such that \( MP = 0 \) as follows:

1. Pick \( r, s \in \mathbb{Z}_{>0} \) such that \( rs \geq 4\sqrt{q} \) and put \( a := \lceil (\sqrt{q} - 1)^2 \rceil = \min(\mathcal{H}(q) \cap \mathbb{Z}) \).

2. Compute **baby steps** \( S_{\text{baby}} := \{0, P, 2P, \ldots, (r - 1)P\} \).

3. Compute **giant steps** \( S_{\text{giant}} := \{aP, (a + r)P, (a + 2r)P, \ldots, (a + (s - 1)r)P\} \).

4. For each \( P_{\text{giant}} = (a + ir)P \) check if \( P_{\text{giant}} + P_{\text{baby}} = 0 \) for some \( P_{\text{baby}} = jP \). If so, output \( M = a + ri + j \).

Every \( M \in \mathcal{H}(q) \) can be written as \( M = a + ir + j \) with \( 0 \leq i < s \) and \( 0 \leq j < r \), and

\[
MP = (a + ri)P + jP = P_{\text{giant}} + P_{\text{baby}} = 0,
\]

for some \( P_{\text{giant}} \in S_{\text{giant}} \) and \( P_{\text{baby}} \in S_{\text{baby}} \). **Complexity** is \( O(\exp(n/4)M(n)) \).
Batching inversions

In order to efficiently match giant steps with baby steps we use affine coordinates. Addition in $E(\mathbb{F}_q)$ uses $3M + I$ or $4M + I$ operations in $\mathbb{F}_q$, or $O(M(n) \log n)$ time.

Algorithm

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 compute $\beta_i := \beta_{i-1} \alpha_i$ for $i$ from 1 to $m$.
2. Compute $\gamma_m := \beta_m^{-1}$.
3. For $i$ from $m$ down to 1 compute $\alpha_i^{-1} := \beta_{i-1} \gamma_i$ and $\gamma_{i-1} := \gamma_i \alpha_i$.

This takes less than $3mM + I$ operations in $\mathbb{F}_q$, or $O(mM(n) + M(n) \log n)$ time. For $m \geq \log n$ this is $O(M(n))$ per inversion, on average, rather than $O(M(n) \log n)$. For large $m$ the cost of each baby/giant step is effectively $6M$ operations in $\mathbb{F}_q$. 
The table below summarizes the complexity of various algorithms to compute $\#E(\mathbb{F}_q)$. Complexity bounds are bit-complexities in terms of $n = \log q$.

<table>
<thead>
<tr>
<th>algorithm</th>
<th>time complexity</th>
<th>space complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Totally naive</td>
<td>$O(\exp(2n)M(n))$</td>
<td>$O(n)$</td>
</tr>
<tr>
<td>Legendre symbols on the fly</td>
<td>$O(\exp(n)M(n) \log n)$</td>
<td>$O(n)$</td>
</tr>
<tr>
<td>Legendre symbols precomputed</td>
<td>$O(\exp(n)M(n))$</td>
<td>$O(\exp(n)n)$</td>
</tr>
<tr>
<td>Mestre with linear search</td>
<td>$O(\exp(n/2)M(n))$</td>
<td>$O(n)$</td>
</tr>
<tr>
<td>Mestre with baby-steps giant-steps</td>
<td>$O(\exp(n/4)M(n))$</td>
<td>$O(\exp(n/4)n)$</td>
</tr>
<tr>
<td>Schoof’s algorithm</td>
<td>$O(\text{poly}(n))$</td>
<td>$O(\text{poly}(n))$</td>
</tr>
</tbody>
</table>

For Mestre’s algorithm these are expected running times, the rest are deterministic. Probabilistic optimizations to Schoof’s algorithm (SEA) are used in practice for large $q$. 