The point counting problem

Graeme Taylor

Edinburgh

June 11, 2007

The point Counting Problem

The group law on elliptic curves is well-known and gives rise to elliptic curve cryptography systems which find application to government and industry today. However, the generalisation to higher genus requires the manipulation of divisor classes rather than points, and analogues of key genus 1 results have yet to be found. Nonetheless, effective computation within the group is possible, and techniques for finding the cardinality of hyperelliptic curve jacobians are improving, with calculations over curves of cryptographically significant size having recently been achieved. This report sets out the mathematical background to this problem, its cryptographic application and a solution strategy from algebraic geometry; and discusses the two main approaches employed in its attack.

Curves and Points

Definition

Let $f \in K[u]$ be a squarefree monic polynomial of degree $2g+1$ and $h \in K[u]$ be of degree at most g. Then a curve with affine model

$$C : v^2 - h(u)v = f(u)$$

is described as a *hyperelliptic curve of genus g*. The special case of genus 1, gives an *elliptic curve*.

Definition

A pair $P = (x, y) \in \bar{K} \times \bar{K}$ is described as a *point of C* if $y^2 - h(x)y = f(x)$. The point is *rational* if $(x, y) \in K \times K$.
Since we work in affine rather than projective space, there is also the *rational point at infinity*, ∞.

Divisors

Definition

A *divisor* D of C is a finite formal sum of points of C:

$$D = \sum_i{}' m_i P_i \quad m_i \in \mathbb{Z}$$

Its *degree* is given by $\sum_i m_i$.

Definition

The *group of divisors* Div_C is the set of divisors equipped with formal (pointwise) addition; it has a subgroup, Div^0_C, consisting of the degree 0 divisors.

Principal Divisors

Definition

Consider a function $h \in K(C)$ denoted $h = p/q$ for $p, q \in K[u, v]$ such that $v^2 - f \not| q$: that is, q is not everywhere zero on C. Then h will have a finite set of zeros (those of p) and of poles (zeros of q); we associate to h a divisor, (h), where the P_i are those zeros and poles and m_i their multiplicities:

$$(h) := \mathrm{div(h)}_0 - \mathrm{div(h)}_\infty = \sum_{\substack{P_i \in \\ \{zeros\ of\ p\}}} ord_{P_i}(p) P_i - \sum_{\substack{P_i \in \\ \{zeros\ of\ q\}}} ord_{P_i}(q) P_i$$

Definition

If there is a nonzero function h on C such that D a divisor is (h), then D is described as *principal*.

Jacobian

The set Princ_C of principal divisors is a subgroup of Div_C^0.

Definition

The *divisor class group of C of degree zero* or *Picard group of C*, (equivalently here the *Jacobian of C*) is the quotient group

$$\mathrm{Pic}_C^0 = \mathrm{Div}_C^0 / \mathrm{Princ}_C$$

Thus D_1, D_2 are in the same class if $\exists f \in K(C)$ s.t. $\mathrm{div}(f) = D_1 - D_2$.

Any divisor $D \in \mathrm{Div}_C^0$ will have a representative of *weight r*

$$D = \sum_{i=1}^{r} P_i - r\infty$$

such that if P_i is a point in the sum, then $P_j \neq -P_i$ for any $j \neq i$. Such a representation is called *semi-reduced*.
If $r \leq g$ the representation is called *reduced*. By Riemann-Roch, any divisor class in Pic_C^0 has a reduced representative.

Mumford polynomials

Definition

Let D be a semi-reduced divisor whose points are $P_i = (x_i, y_i)$. We associate to D polynomials $a, b \in \bar{K}[u]$ such that

$$a(u) = \prod_i^r (u - x_i)$$

$$b(x_i) = y_i \; 1 \leq i \leq r$$

such that b has degree less than that of a, and the appropriate multiplicity for repeated points- i.e., if P_i occurs k times in the semi-reduced representation of D, then $(u - x_i)^k$ divides $b - y_i$. We write

$$D = div(a, b)$$

Definition

$D = div(a, b)$ is *rational* if $a, b \in K[u]$.

Group Law (roughly!)

- Given divisor classes $\mathcal{D}_1, \mathcal{D}_2 \in \mathcal{J}_K(C)$, take rational reduced representatives D_1, D_2.
- Form a new rational, semi-reduced divisor $D_1 + D_2$ by combining the points of D_1, D_2 in Div^0_C.
- Reduce modulo Princ_C to some rational D of degree at most g.
- Define $\mathcal{D}_1 \oplus \mathcal{D}_2$ to be the equivalence class of D in $\mathcal{J}_K(C)$.

Explicit algorithms have been developed for these; see report for description or website for SAGE and Maple implementations.

DLP

Let $(G, \oplus)$ be an additive cyclic group of prime order p generated by an element g. We can define a map

$$\varphi : \mathbb{Z} \to G$$

$$n \mapsto [n]g = \underbrace{g \oplus g \oplus \cdots \oplus g}_{n \text{ copies}}$$

This gives an isomorphism between $(\mathbb{Z}/p\mathbb{Z}, +)$ and $(G, \oplus)$. Given $g, h \in G$, the *Discrete Logarithm Problem (DLP)* is to find $k \in \mathbb{Z}$ such that $[k]g = h$.

DLP

Example

Let $g = a + p\mathbb{Z}$ be a generator of $(\mathbb{Z}/p\mathbb{Z}, +)$, and $h = b + p\mathbb{Z}$ another element. Then the DLP

$$[k]g = h$$

has solution

$$k = a^{-1}b \pmod{p}$$

and this calculation is of polynomial complexity in p.

For secure DLP cryptography we need a cyclic group such that computation of the group law is efficient, but the isomorphism with $\mathbb{Z}/p\mathbb{Z}$ is not apparent from the group elements. It is believed that the groups of rational points on elliptic curves / rational divisors on hyperelliptic curves are indeed suitable.

Frobenius Endomorphism

An alternative characterisation of rational divisors can be obtained via the Frobenius endomorphism:

Definition

The *Frobenius morphism* is the map $\phi_q : \alpha \mapsto \alpha^q$. It extends naturally to points of $\bar{K}$; to polynomials over $\bar{K}$ coefficient-wise, and hence to divisors $div(a, b)$, leading to π, the *Frobenius endomorphism* of Pic^0_C.

Characteristic Polynomial of Frobenius

A polynomial is fixed by π if and only if its coefficients are from K; hence $\mathcal{J}_K(C) = ker(id_{\mathrm{Pic}^0_C} - \pi)$ and so

$$\#\mathcal{J}_K(C) = deg(id_{\mathrm{Pic}^0_C} - \pi)$$

π acts linearly as an element of $\mathrm{End}(\mathrm{Pic}^0_C)$: denoting its characteristic polynomial as $\chi(T)$, we have

$$\#\mathcal{J}_K(C) = \chi(1)$$

Theorem

(Weil Theorems)
$\chi(T)$ is a monic integer polynomial of degree $2g$ with roots λ_i (the eigenvalues of Frobenius) of absolute value $\sqrt{q}$.

Weil Interval

Corollary

(Weil Interval)

$$(\sqrt{q}-1)^{2g} \leq \#\mathcal{J}_K(C) \leq (\sqrt{q}+1)^{2g}$$

Example

For a hyperelliptic curve of genus 2, we have:

- $\chi(T) = T^4 - s_1 T^3 + s_2 T^2 - q s_1 T + q^2$
- $\#\mathcal{J}_K(C) = 1 - s_1 + s_2 - q s_1 + q^2 \leq (\sqrt{q}+1)^4$
- $|s_1| \leq 4\sqrt{q}$, $|s_2| \leq 6q$.

Thus in genus 2 it suffices to compute $\mathcal{J}_K(C)$ modulo $w = (\sqrt{q}+1)^4 - (\sqrt{q}-1)^4 = 2\lfloor 4(q+1)\sqrt{q} \rfloor$.

Generic Algorithms

Theorem

Generic Group order algorithm
INPUT: A group G and interval $[a, b]$ of width w s.t. $|G| \in [a, b]$.
OUTPUT: The order of G.

1. *Generate a random element $g \in G$.*
2. *compute n=$ord(g)$ and set $e = n$.*
3. *While $e < w$:*
   Generate a random element $g \in G$.
   set $e = lcm\{e, ord(g)\}$
4. *Return the unique $N \in [a, b]$ s.t. $N \equiv 0$ mod e.*

Generic Algorithms

- e converges to the exponent of the group, which necessarily divides its order N (so $N \equiv 0$ modulo e).
- Thus algorithm will fail to terminate for groups where the exponent is less than w.
- Algorithm requires only a "black box" for the group law and random element generation but depends upon the calculation of the order of such elements.
- This is a special case of the discrete logarithm problem!
- The best generic algorithms for the DLP of recovering n such that $[n]g = h$ for some $g, h \in G$ - the *Baby Steps, Giant Steps algorithm* and *Pollard's Rho method* - have complexity of order $\sqrt{n}$, and are thus known as square-root algorithms.
- It was conjectured that a similar complexity bound would hold for order calculation, but a very recent result (Sutherland, 2007) shows that a lower bound holds there.

Schoof's Algorithm

Since the cardinality of $\mathcal{J}_K(C)$ and the coefficients of $\chi(T)$ are bounded, their exact values can be recovered from their values modulo a selection of primes via the chinese remainder theorem. For l coprime to p, the characteristic polynomial of π modulo l is the characteristic polynomial of π restricted to the l-torsion subgroup.

Elliptic Curves

In the genus 1 case, the l-torsion elements are characterised by the division polynomials ϕ_l.
χ takes a particularly simple form, such that we need only determine its *trace* t satisfying

$$\pi^2 - [t]\pi + [q] = [0]$$

Schoof's original approach to step 2 is to recover t_l by brute force determination of a $\tau \in 1, \ldots l-1$ such that

$$(x^{q^2}, y^{q^2}) \oplus [q_l](x, y) = [\tau](x^q, y^q)$$

There are explicit formulae for the multiplication-by-m isogeny, and intermediate expressions are controlled by working modulo the l-division ideal generated by ϕ_l (with coefficients further constrained modulo l).
This algorithm is then of polynomial time complexity, but is still too slow for curve sizes of cryptographic interest: the ϕ_l are of degree $(l^2-1)/2$ and hence impractical beyond $q \approx 10^{200}$.

Schoof's Algorithm

Theorem

Schoof's algorithm in genus 1

INPUT: Curve $E/\mathbb{F}_q$

OUTPUT: $\#E(\mathbb{F}_q)$ *the cardinality of* E.

1. *Compute* L *a set of primes such that*

$$\prod_{l \in L} l \geq 4\sqrt{q} \tag{1}$$

   with L *minimal*
2. *For each* $l \in L$, *compute* t_l, *the trace modulo* l.
3. *By the Chinese Remainder theorem, find* t_L, *the trace modulo* $\prod_{l \in L} l$.
4. *Expressing* t_L *as* t *in the range* $-2\sqrt{q} \leq t \leq 2\sqrt{q}$ *gives the true trace.*
5. *Return* $q + 1 - t$.

Refinements: SEA

Refinements by Elkies and Atkin give rise to the SEA algorithm: by way of modular polynomials, primes l can be characterised as either Elkies or Atkin type.

- For Elkies primes, ϕ_l can be replaced by a factor of degree $(l-1)/2$.
- Atkin primes give rise to an easily computed set of candidates for t_l and hence t which can be tested against random points, although the size of this set grows exponentially and thus careful choice of primes in stage 1 is necessary.
- Without precomputation, the determination of the modular polynomials can be harder than naive determination of t_l.
- Nonetheless, the SEA algorithm introduces significant performance gains and is effective for $q \approx 10^{500}$.
- Record (Nov 2006): computation over $\mathbb{F}_p$ with $p = 10^{2499} + 7131$ was completed (although this took over a year).

Hyperelliptic Curves

Problems

- Have to work with l-division ideals which may not be principal.
- Although hyperelliptic analogues of the modular polynomials have been developed, they have not lead to an Elkies procedure.

Schoof-like algorithm in genus 2

Theorem

Schoof's algorithm in genus 2

INPUT: Curve $C/\mathbb{F}_q$

OUTPUT: $\#\mathcal{J}_K(C)$.

1. *For sufficiently many small primes* l:
   Set $L = \{(s_1, s_2); s_1, s_2 \in [0, l-1]\}$.
   While $\#L > 1$ *do:*
   - *Construct an* l*-torsion divisor* D
   - *Eliminate elements of* L *such that*

$$\pi^4(D) - s_1\pi^3(D) + s_2\pi^2(D) - (qs_1 \bmod l)\pi(D) + (q^2 \bmod l)D \neq 0$$

   - *Deduce* $\chi(T) \bmod l$ *from the final pair* s_1, s_2.
2. *Construct* $\chi(T)$ *from the* $\chi(T) \bmod l$ *by the chinese remainder theorem.*
3. *Return* $\chi(1)$

Hybrid Approaches

Theorem

Gaudry-Harley point counting algorithm

INPUT: Curve $C/\mathbb{F}_q$ of genus 2.
OUTPUT: $\#\mathcal{J}_K(C)$.

1. *Compute $\#\mathcal{J}_K(C)$ mod 2^e by the halving algorithm.*
2. *For primes $l = 2, 3, 5, \ldots, l_{max}$:*
   - *Compute $\chi(T)$ mod l by a Schoof-like algorithm.*
   - *Compute $\#\mathcal{J}_K(C)$ mod l from $\chi(T)$ mod l.*
3. *Compute $\chi(T)$ mod p via the Cartier-Manin operator.*
4. *Compute $\#\mathcal{J}_K(C)$ mod p from $\chi(T)$ mod p.*
5. *Compute $\#\mathcal{J}_K(C)$ mod $m = 2^e \cdot 3 \cdots l_{max} \cdot p$ by CRT.*
6. *Compute $\#\mathcal{J}_K(C)$ by a square root algorithm that exploits knowledge of $\#\mathcal{J}_K(C)$ mod m.*