Hyperelliptic Curves over Finite Fields

Graeme Taylor

Edinburgh

April 27, 2007

The Discrete Logarithm Problem

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

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

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

This gives an isomorphism between $(\mathbb{Z}/p\mathbb{Z}, +)$ and $(G, \oplus)$. The *discrete logarithm problem to base g (DLP)* is to compute the inverse map:

The Discrete Logarithm Problem

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

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

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

This gives an isomorphism between $(\mathbb{Z}/p\mathbb{Z}, +)$ and $(G, \oplus)$. The *discrete logarithm problem to base g (DLP)* is to compute the inverse map:

Discrete Logarithm Problem

Given $g, h \in G$, find $k \in \mathbb{Z}$ such that $[k]g = h$.

The Discrete Logarithm Problem

The DLP needn't be difficult!

If the isomorphism between G and $\mathbb{Z}/p\mathbb{Z}$ is obvious, then the DLP is easy, since it is easy in $(\mathbb{Z}/p\mathbb{Z}, +)$

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.

ElGamal public key encryption

- Let g generate G and suppose Bob has private key a; letting $h = [a]g$ he can safely release as a public key (g, h) as (assuming hard DLP) there's no easy way to retrieve a.

ElGamal public key encryption

- Let g generate G and suppose Bob has private key a; letting $h = [a]g$ he can safely release as a public key (g, h) as (assuming hard DLP) there's no easy way to retrieve a.
- Suppose Alice wishes to send a message $m \in G$. She picks some $k \in 1 \ldots, |G|$ randomly and computes $\gamma = [k]g$, $\delta = m \oplus [k]h$.

ElGamal public key encryption

- Let g generate G and suppose Bob has private key a; letting $h = [a]g$ he can safely release as a public key (g, h) as (assuming hard DLP) there's no easy way to retrieve a.
- Suppose Alice wishes to send a message $m \in G$. She picks some $k \in 1 \ldots, |G|$ randomly and computes $\gamma = [k]g$, $\delta = m \oplus [k]h$.
- Bob receives the ciphertext $c = (\gamma, \delta)$. He, unlike everyone else, knows a so can find $[a]\gamma = [ak]g$.
- Finding the inverse is easy; call that d.

ElGamal public key encryption

- Let g generate G and suppose Bob has private key a; letting $h = [a]g$ he can safely release as a public key (g, h) as (assuming hard DLP) there's no easy way to retrieve a.
- Suppose Alice wishes to send a message $m \in G$. She picks some $k \in 1 \ldots, |G|$ randomly and computes $\gamma = [k]g$, $\delta = m \oplus [k]h$.
- Bob receives the ciphertext $c = (\gamma, \delta)$. He, unlike everyone else, knows a so can find $[a]\gamma = [ak]g$.
- Finding the inverse is easy; call that d.
- Then $d \oplus \delta = m$, Alice's message.

Groups of rational divisors

Requirements

For secure DLP cryptography we need a cyclic group such that computation 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 suitable.

Groups of rational divisors

Requirements

For secure DLP cryptography we need a cyclic group such that computation 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 suitable.

Strategy/Requirements

- Give a representation of the group of rational divisors and their group law.
- Compute the cardinality of the group. *(this step is also of number-theoretic interest.)*
- Identify a large prime subgroup G and a generator g to use for ElGamal. *(this issue will not be covered.)*

Curves

Monster-barring

Throughout, we assume a field $K = \mathbb{F}_q$ of characteristic p not 2 or 3, with algebraic closure A; and a non-singular curve, considered over affine space. Then we can work with the following:

Curves

Monster-barring

Throughout, we assume a field $K = \mathbb{F}_q$ of characteristic p not 2 or 3, with algebraic closure A; and a non-singular curve, considered over affine space. Then we can work with the following:

Definition

$C : v^2 = f(u)$ defines a *hyperelliptic curve of genus* g over K if $f \in K[u]$ is of degree $2g+1$ with distinct roots in K.

Points and Divisors

Definition

A pair $P = (x, y) \in A \times A$ is described as a *point of* C if $y^2 = f(x)$.
Note that P needn't have coefficients from K, merely A!
There is also the *point at infinity*, ∞.

Points and Divisors

Definition

A pair $P = (x, y) \in A \times A$ is described as a *point of C* if $y^2 = f(x)$.
Note that P needn't have coefficients from K, merely A!
There is also the *point at infinity*, ∞.

Definition

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

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

Points and Divisors

Definition

The *degree* or *weight* of D is $\sum_i m_i$.

Points and Divisors

Definition

The *degree* or *weight* of D is $\sum_i m_i$.

Definition

The *group of divisors* $\mathcal{D}$ is the set of divisors equipped with formal (pointwise) addition; it has a subgroup, $\mathcal{D}_0$, consisting of the degree 0 divisors.

Functions

Any polynomial $p(u, v)$ can be considered as a function on C of the form $p = a(u) + b(u)v$, since $v^2 = f(u)$.
If p vanishes at (x, y) then the order of the zero (x, y) of p is the exponent of the highest power of $(u - x)$ which divides $a^2 - b^2 f$.

Functions

Any polynomial $p(u, v)$ can be considered as a function on C of the form $p = a(u) + b(u)v$, since $v^2 = f(u)$.
If p vanishes at (x, y) then the order of the zero (x, y) of p is the exponent of the highest power of $(u - x)$ which divides $a^2 - b^2 f$.

Definition

Thus we can define functions on C as $h = p/q$ 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:

$$\sum_{\text{zeros of } p} ord_{P_i}(p) P_i - \sum_{\text{zeros of } q} ord_{P_i}(q) P_i$$

Principal Divisors

Definition

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

Principal Divisors

Definition

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

Definition

The set of principal divisors, $\mathcal{P}$, is actually a subgroup of $\mathcal{D}_0$ and hence of $\mathcal{D}$.

Principal Divisors

Definition

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

Definition

The set of principal divisors, $\mathcal{P}$, is actually a subgroup of $\mathcal{D}_0$ and hence of $\mathcal{D}$.

Definition

The *Jacobian of* C, $\mathcal{J}$, is the quotient group $\mathcal{D}_0/\mathcal{P}$.

Reduced Divisors

Any $D \in \mathcal{J}$ will have a representation

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

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*.

Reduced Divisors

Any $D \in \mathcal{J}$ will have a representation

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

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*.

Theorem

Any $D \in \mathcal{J}$ has a reduced representation.
(this follows from Riemann-Roch)

Group Law (roughly!)

- Take divisors D_1, D_2 in reduced form
- Form a new, semi-reduced divisor $D_1 + D_2$ by combining the points of D_1, D_2
- Reduce to some D of degree at most g, this is $D_1 \oplus D_2$

Group Law (roughly!)

- Take divisors D_1, D_2 in reduced form
- Form a new, semi-reduced divisor $D_1 + D_2$ by combining the points of D_1, D_2
- Reduce to some D of degree at most g, this is $D_1 \oplus D_2$

Notice that for genus 1 (elliptic curves) we are combining a pair of points into a point: this is the usual chord-and-tangent process. But for higher genus, the sum of two points needn't be a point, as the divisor consisting of their sum needn't reduce further.

Group Law (roughly!)

- Take divisors D_1, D_2 in reduced form
- Form a new, semi-reduced divisor $D_1 + D_2$ by combining the points of D_1, D_2
- Reduce to some D of degree at most g, this is $D_1 \oplus D_2$

Notice that for genus 1 (elliptic curves) we are combining a pair of points into a point: this is the usual chord-and-tangent process. But for higher genus, the sum of two points needn't be a point, as the divisor consisting of their sum needn't reduce further.

So the set of rational points don't form a subgroup! Worse, as it stands we don't actually have a description of rationality; nor an explicit description of the reduction process. Computing with the constituent points is also undesirable.

Mumford Polynomial representation of Divisors

Definition

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

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

Rational Divisors

Definition

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

Rational Divisors

Definition

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

Beware that a rational divisor needn't be the sum of rational points! However, this will be true of a weight 1 rational divisor; which is a single rational point thought of as a divisor.

Rational Divisors

Definition

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

Beware that a rational divisor needn't be the sum of rational points! However, this will be true of a weight 1 rational divisor; which is a single rational point thought of as a divisor.
For an elliptic curve, therefore, the rational divisors *are* isomorphic to the rational points of the curve; this fails in higher genus.

G, at last

- The set $\mathcal{J}_K$ of rational elements of the Jacobian is a subgroup.
- We can work over $K[u]$ with the polynomials a, b; clever manipulation of gcds allows for the group law computation without decomposing into the points (which would often mean working in A).
- There are explicit formulae for composition and reduction (see website for details and Maple implementation).

G, at last

- The set $\mathcal{J}_K$ of rational elements of the Jacobian is a subgroup.
- We can work over $K[u]$ with the polynomials a, b; clever manipulation of gcds allows for the group law computation without decomposing into the points (which would often mean working in A).
- There are explicit formulae for composition and reduction (see website for details and Maple implementation).

So we can compute in $\mathcal{J}_K$; a large prime subgroup would be suitable for use as G in an ElGamal cryptosystem. For this, we need $\#\mathcal{J}_K$: this is the point counting problem.

Frobenius endomorphism

Definition

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

Frobenius endomorphism

Definition

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

Theorem

For $K = \mathbb{F}_q$

- $\mathcal{J}_K = ker(id_{\mathcal{J}} - \Phi_q)$.
- *Hence* $\#\mathcal{J}_K = deg(id_{\mathcal{J}} - \phi_q)$.

Characteristic Polynomial of Frobenius

Associated to ϕ_q is a polynomial $\chi(T)$, the *characteristic polynomial of Frobenius*. The definition is in terms of l-adic Galois representation; we don't need it! We call the roots λ_i of χ the *eigenvalues of the Frobenius endomorphism.*

Characteristic Polynomial of Frobenius

Associated to ϕ_q is a polynomial $\chi(T)$, the *characteristic polynomial of Frobenius*. The definition is in terms of l-adic Galois representation; we don't need it! We call the roots λ_i of χ the *eigenvalues of the Frobenius endomorphism.*

- χ is monic, degree $2g$ and has integer coefficients.
- $\#\mathcal{J}_K = \chi(1)$
- *Riemann Hypothesis for curves over finite fields*: $|\lambda_i| = \sqrt{q}$.
- The characteristic polynomial for the restriction of the Frobenius endomorphism to $\mathcal{J}[n]$ is χ mod n.

Zeta Functions

Definition

For X an algebraic variety over $\mathbb{F}_q$, let N_k be the number of $\mathbb{F}_{q^k}$-rational points on X.
Then the *zeta function of* X *over* $\mathbb{F}_q$ is

$$Z(T) := \exp\left(\sum_{k=1}^{\infty} N_k \frac{T^k}{k}\right)$$

Zeta Functions

Definition

For X an algebraic variety over $\mathbb{F}_q$, let N_k be the number of $\mathbb{F}_{q^k}$-rational points on X.
Then the *zeta function of X over $\mathbb{F}_q$* is

$$Z(T) := \exp\left(\sum_{k=1}^{\infty} N_k \frac{T^k}{k}\right)$$

Theorem

For C a smooth genus g projective curve we have

$$Z(T) = \frac{T^{2g}\chi(1/T)}{(1-T)(1-qT)} = \frac{\prod_{i=1}^{2g}(1-\lambda_i T)}{(1-T)(1-qT)}$$

Weil Intervals

From the previous observations, we have for C a genus g hyperelliptic curve defined over $K = \mathbb{F}_q$:

Theorem

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

Weil Intervals

From the previous observations, we have for C a genus g hyperelliptic curve defined over $K = \mathbb{F}_q$:

Theorem

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

Theorem

$$N_k = q^k + 1 - \sum_{i=1}^{2g} \lambda_i^k$$

Hence we have the Hasse-Weil bound

$$-2g\sqrt{q} + q + 1 \leq \#C(\mathbb{F}_q) \leq 2g\sqrt{q} + q + 1$$

Interval searches

These two sets of bounds coincide for elliptic curves, since then the set of rational points on the curve is identified with the Jacobian. For hyperelliptic curves, knowledge of $\#C(\mathbb{F}_q)$ is of interest from a number-theory point of view, but as the set is not a group it is $\#\mathcal{J}_K$ that we need for cryptographic purposes.

Interval searches

These two sets of bounds coincide for elliptic curves, since then the set of rational points on the curve is identified with the Jacobian. For hyperelliptic curves, knowledge of $\#C(\mathbb{F}_q)$ is of interest from a number-theory point of view, but as the set is not a group it is $\#\mathcal{J}_K$ that we need for cryptographic purposes.

Let w be the width of the interval; then it suffices to determine $\#\mathcal{J}_K$ mod w.
This motivates a number of techniques; these can often be combined by the Chinese Remainder Theorem (if we know $\#\mathcal{J}_K$ modulo coprime values $p_1 \ldots, p_n$ then we know it modulo $p_1 \times \cdots \times p_n$).

Element Orders

- It's fairly easy to generate random rational points of C and hence to form random divisors from $\mathcal{J}_K$.

Element Orders

- It's fairly easy to generate random rational points of C and hence to form random divisors from $\mathcal{J}_K$.
- If D is such a divisor, then the order of D divides $\#\mathcal{J}_K$, so $\#\mathcal{J}_K \equiv 0 \bmod ord(D)$.

Element Orders

- It's fairly easy to generate random rational points of C and hence to form random divisors from $\mathcal{J}_K$.
- If D is such a divisor, then the order of D divides $\#\mathcal{J}_K$, so $\#\mathcal{J}_K \equiv 0 \bmod ord(D)$.
- Generating several such D and finding the lcm of their orders will converge to the group exponent, often (but not always) greater than w.

Element Orders

- It's fairly easy to generate random rational points of C and hence to form random divisors from $\mathcal{J}_K$.
- If D is such a divisor, then the order of D divides $\#\mathcal{J}_K$, so $\#\mathcal{J}_K \equiv 0 \bmod ord(D)$.
- Generating several such D and finding the lcm of their orders will converge to the group exponent, often (but not always) greater than w.
- But, finding the order of the element corresponds to finding the least n such that $[n]D = id$: this is a discrete logarithm problem!

Element Orders

- It's fairly easy to generate random rational points of C and hence to form random divisors from $\mathcal{J}_K$.
- If D is such a divisor, then the order of D divides $\#\mathcal{J}_K$, so $\#\mathcal{J}_K \equiv 0 \bmod ord(D)$.
- Generating several such D and finding the lcm of their orders will converge to the group exponent, often (but not always) greater than w.
- But, finding the order of the element corresponds to finding the least n such that $[n]D = id$: this is a discrete logarithm problem!
- Generic algorithms (such as Baby Step Giant Step) take at best $O(\sqrt{n})$ group operations to determine an element order n.

Element Orders

- It's fairly easy to generate random rational points of C and hence to form random divisors from $\mathcal{J}_K$.
- If D is such a divisor, then the order of D divides $\#\mathcal{J}_K$, so $\#\mathcal{J}_K \equiv 0 \bmod ord(D)$.
- Generating several such D and finding the lcm of their orders will converge to the group exponent, often (but not always) greater than w.
- But, finding the order of the element corresponds to finding the least n such that $[n]D = id$: this is a discrete logarithm problem!
- Generic algorithms (such as Baby Step Giant Step) take at best $O(\sqrt{n})$ group operations to determine an element order n.
- Using BSGS, this approach is suitable for $q \approx 10^{30}$ for an elliptic curve.

Schoof's algorithm on Elliptic curves

In genus 1, χ takes a particularly simple form, and we need only determine its trace t, which satisfies

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

Schoof's algorithm on Elliptic curves

In genus 1, χ takes a particularly simple form, and we need only determine its trace t, which satisfies

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

Recall that we can work in the l-torsion subgroup and study χ mod l instead; then we can test by brute force $\tau \in 1, \ldots, l-1$ for

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

This will give t_l for assorted primes l.

Schoof's algorithm

Schoof's algorithm

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

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

1. *Compute L a set of primes such that*
$$\prod_{l \in L} l \geq 4\sqrt{q} \qquad (1)$$
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 $q + 1 - t$.*

Improving step 2?

- Schoof's algorithm is of polynomial time complexity, so practical in theory.

Improving step 2?

- Schoof's algorithm is of polynomial time complexity, so practical in theory.
- The intermediate expressions are controlled by working modulo the lth division polynomial (of degree $(l^2 - 1)/2$) and the curve equation.

Improving step 2?

- Schoof's algorithm is of polynomial time complexity, so practical in theory.
- The intermediate expressions are controlled by working modulo the lth division polynomial (of degree $(l^2 - 1)/2$) and the curve equation.
- In practice, though, it is still too slow for curve sizes of cryptographic interest: No use over fields beyond approx 10^{200} elements, where we'd need $l \approx 250$.

Improving step 2?

- Schoof's algorithm is of polynomial time complexity, so practical in theory.
- The intermediate expressions are controlled by working modulo the lth division polynomial (of degree $(l^2-1)/2$) and the curve equation.
- In practice, though, it is still too slow for curve sizes of cryptographic interest: No use over fields beyond approx 10^{200} elements, where we'd need $l \approx 250$.
- Improvements to step 2 by Elkies and Atkin give rise to the SEA algorithm, this is effective for $q \approx 10^{500}$.

Improving step 2?

- Schoof's algorithm is of polynomial time complexity, so practical in theory.
- The intermediate expressions are controlled by working modulo the lth division polynomial (of degree $(l^2 - 1)/2$) and the curve equation.
- In practice, though, it is still too slow for curve sizes of cryptographic interest: No use over fields beyond approx 10^{200} elements, where we'd need $l \approx 250$.
- Improvements to step 2 by Elkies and Atkin give rise to the SEA algorithm, this is effective for $q \approx 10^{500}$.
- Record (November 2006) is $p = 10^{2499} + 7131$, although this took over a year to complete!

Improving step 2

We characterise primes l as either Elkies or Atkin primes:

Definition

If

$$F_l = u^2 - t_l u + q_l = (u - \lambda)(u - \mu)$$

then l is an Elkies prime iff $\lambda, \mu \in \mathbb{F}_l$.

Of course, if we knew the factorisation, then we'd already know t_l!

Improving step 2

We characterise primes l as either Elkies or Atkin primes:

Definition

If

$$F_l = u^2 - t_l u + q_l = (u - \lambda)(u - \mu)$$

then l is an Elkies prime iff $\lambda, \mu \in \mathbb{F}_l$.

Of course, if we knew the factorisation, then we'd already know t_l!

Definition

For a curve E with j-invariant j and a prime l, the *modular polynomial of order* l is the polynomial of degree $l + 1$ whose roots are the j-invariants of the curves isogeneous to E such that the kernel of the isogeny is of size l.

These are hard to compute! But their splitting type tells us whether l is an Elkies or Atkin prime.

Elkies and Atkin procedures

Elkies primes

Elkies describes a procedure for replacing the lth division polynomial with a factor of degree $(l-1)/2$; this allows for much faster computation in practice (despite the same theoretical complexity - polynomial time - as Schoof). Further, we need only find a $\lambda \in 1, \ldots, l-1$ by trial and error such that

$$(x^q, y^q) = [\lambda](x, y)$$

as then λ is a root of F_l and $t_l = \lambda + q/\lambda \bmod l$.

Elkies and Atkin procedures

Atkin primes

Atkin gives a separate procedure for finding t_l for l a non-Elkies prime: actually, it gives a set of candidates for t_l, which must be tested against random points once combined with Elkies data. This is of exponential complexity but is computationally simple and thus often helpful in practice.

We do this by noting that $\lambda, \mu \in \mathbb{F}_{l^2} \backslash \mathbb{F}_l$ and $\gamma_r = \lambda/\mu$ is an element of known order r in $\mathbb{F}_{l^2}$; there are only finitely many possibilities for γ_r and thus for t_l.

Higher genus

- SEA only works for genus 1 at present.

Higher genus

- SEA only works for genus 1 at present.
- Modular polynomials for higher genus have been developed, but there is no equivalent of the Elkies procedure.

Higher genus

- SEA only works for genus 1 at present.
- Modular polynomials for higher genus have been developed, but there is no equivalent of the Elkies procedure.
- The schoof procedure itself is more difficult, since χ, not just its trace, must be found.

Higher genus

- SEA only works for genus 1 at present.
- Modular polynomials for higher genus have been developed, but there is no equivalent of the Elkies procedure.
- The schoof procedure itself is more difficult, since χ, not just its trace, must be found.
- Working with $\mathcal{J}_K[l]$ becomes increasingly difficult as g grows: e.g., need to work modulo an ideal rather than a single division polynomial.

Higher genus

- SEA only works for genus 1 at present.
- Modular polynomials for higher genus have been developed, but there is no equivalent of the Elkies procedure.
- The schoof procedure itself is more difficult, since χ, not just its trace, must be found.
- Working with $\mathcal{J}_K[l]$ becomes increasingly difficult as g grows: e.g., need to work modulo an ideal rather than a single division polynomial.
- But hyperelliptic curves give larger rational jacobians relative to q than elliptic curves, so can work over smaller ground fields yet achieve comparable cryptographic strength.

Thanks!

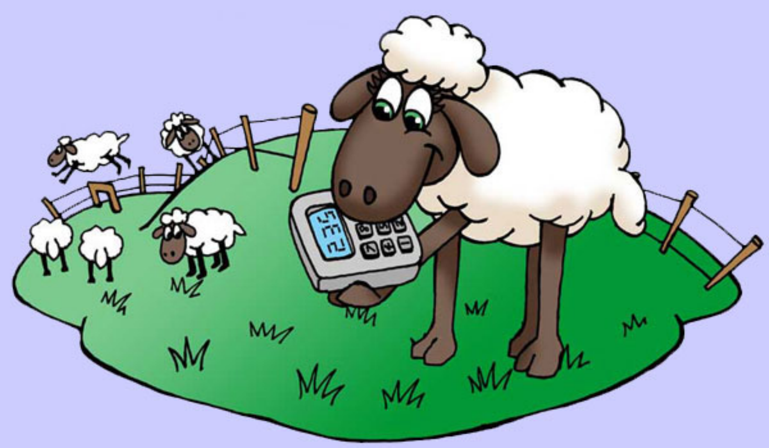

Figure: Point counting in a finite field

Website: http://maths.straylight.co.uk