Estimating the Size of the Image of Deterministic Hash Functions to Elliptic Curves

Pierre-Alain Fouque Mehdi Tibouchi

École normale supérieure

Latincrypt, 2010-08-09

Outline

Outline

Elliptic curve cryptography

- F finite field of characteristic > 3 (for simplicity's sake).
- Recall that an elliptic curve over F is the set of points $(x, y) \in F^2$ such that:

$$y^2 = x^3 + ax + b$$

(with $a, b \in F$ fixed parameters), together with a point at infinity.
- This set of points forms an abelian group where the Discrete Logarithm Problem and Diffie-Hellman-type problems are believed to be hard (no attack better than the generic ones).
- Interesting for cryptography: for k bits of security, one can use elliptic curve groups of order $\approx 2^{2k}$, keys of length $\approx 2k$. Also come with rich structures such as pairings.

Elliptic curve cryptography

- F finite field of characteristic > 3 (for simplicity's sake).
- Recall that an elliptic curve over F is the set of points $(x, y) \in F^2$ such that:
$$y^2 = x^3 + ax + b$$
(with $a, b \in F$ fixed parameters), together with a point at infinity.
- This set of points forms an abelian group where the Discrete Logarithm Problem and Diffie-Hellman-type problems are believed to be hard (no attack better than the generic ones).
- Interesting for cryptography: for k bits of security, one can use elliptic curve groups of order $\approx 2^{2k}$, keys of length $\approx 2k$. Also come with rich structures such as pairings.

Elliptic curve cryptography

- F finite field of characteristic > 3 (for simplicity's sake).
- Recall that an elliptic curve over F is the set of points $(x, y) \in F^2$ such that:

$$y^2 = x^3 + ax + b$$

(with $a, b \in F$ fixed parameters), together with a point at infinity.
- This set of points forms an abelian group where the Discrete Logarithm Problem and Diffie-Hellman-type problems are believed to be hard (no attack better than the generic ones).
- Interesting for cryptography: for k bits of security, one can use elliptic curve groups of order $\approx 2^{2k}$, keys of length $\approx 2k$. Also come with rich structures such as pairings.

Elliptic curve cryptography

- F finite field of characteristic > 3 (for simplicity's sake).
- Recall that an elliptic curve over F is the set of points $(x, y) \in F^2$ such that:
$$y^2 = x^3 + ax + b$$
(with $a, b \in F$ fixed parameters), together with a point at infinity.
- This set of points forms an abelian group where the Discrete Logarithm Problem and Diffie-Hellman-type problems are believed to be hard (no attack better than the generic ones).
- Interesting for cryptography: for k bits of security, one can use elliptic curve groups of order $\approx 2^{2k}$, keys of length $\approx 2k$. Also come with rich structures such as pairings.

Outline

Hashing to elliptic curves is a problem

- Many cryptographic protocols (schemes for encryption, signature, PAKE, IBE, etc.) involve representing a certain numeric value, often a hash value, as an element of the group $\mathbb{G}$ where the computations occur.
- For $\mathbb{G} = \mathbb{Z}_p^*$, simply take the numeric value itself $\bmod p$.
- However, doesn't generalize when $\mathbb{G}$ is an elliptic curve group; e.g. one cannot put the value in the x-coordinate of a curve point, because only about $1/2$ of possible x-values correspond to actual points.
- Elliptic curve-specific protocols have been developed to circumvent this problem (ECDSA for signature, Menezes-Vanstone for encryption, ECMQV for key agreement, etc.), but doing so with all imaginable protocols is unrealistic.

Hashing to elliptic curves is a problem

- Many cryptographic protocols (schemes for encryption, signature, PAKE, IBE, etc.) involve representing a certain numeric value, often a hash value, as an element of the group $\mathbb{G}$ where the computations occur.
- For $\mathbb{G} = \mathbb{Z}_p^*$, simply take the numeric value itself $\bmod p$.
- However, doesn't generalize when $\mathbb{G}$ is an elliptic curve group; e.g. one cannot put the value in the x-coordinate of a curve point, because only about $1/2$ of possible x-values correspond to actual points.
- Elliptic curve-specific protocols have been developed to circumvent this problem (ECDSA for signature, Menezes-Vanstone for encryption, ECMQV for key agreement, etc.), but doing so with all imaginable protocols is unrealistic.

Hashing to elliptic curves is a problem

- Many cryptographic protocols (schemes for encryption, signature, PAKE, IBE, etc.) involve representing a certain numeric value, often a hash value, as an element of the group $\mathbb{G}$ where the computations occur.
- For $\mathbb{G} = \mathbb{Z}_p^*$, simply take the numeric value itself $\bmod p$.
- However, doesn't generalize when $\mathbb{G}$ is an elliptic curve group; e.g. one cannot put the value in the x-coordinate of a curve point, because only about $1/2$ of possible x-values correspond to actual points.
- Elliptic curve-specific protocols have been developed to circumvent this problem (ECDSA for signature, Menezes-Vanstone for encryption, ECMQV for key agreement, etc.), but doing so with all imaginable protocols is unrealistic.

Hashing to elliptic curves is a problem

- Many cryptographic protocols (schemes for encryption, signature, PAKE, IBE, etc.) involve representing a certain numeric value, often a hash value, as an element of the group $\mathbb{G}$ where the computations occur.
- For $\mathbb{G} = \mathbb{Z}_p^*$, simply take the numeric value itself $\bmod p$.
- However, doesn't generalize when $\mathbb{G}$ is an elliptic curve group; e.g. one cannot put the value in the x-coordinate of a curve point, because only about $1/2$ of possible x-values correspond to actual points.
- Elliptic curve-specific protocols have been developed to circumvent this problem (ECDSA for signature, Menezes-Vanstone for encryption, ECMQV for key agreement, etc.), but doing so with all imaginable protocols is unrealistic.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(k^4)$;
  3. security is difficult to analyse.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient: $O(k^4)$;
  3. security is difficult to analyse.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(n^4)$;
  3. security is difficult to analyze.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k - 1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(n^4)$;
  3. security is difficult to analyze.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(n^4)$;
  3. security is difficult to analyze.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(n^4)$;
  3. security is difficult to analyze.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(n^4)$;
  3. security is difficult to analyze.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(n^4)$;
  3. security is difficult to analyze.

The traditional solution

- For k bits of security:
  1. concatenate the hash value with a counter from 0 to $k-1$;
  2. initialize the counter as 0;
  3. if the concatenated value is a valid x-coordinate on the curve, i.e. $x^3 + ax + b$ is a square in F, return one of the two corresponding points; otherwise increment the counter and try again.
- Heuristically, the probability of a concatenated value being valid is $1/2$, so k iterations provide k bits of security.
- However, a number of problems with this solution:
  1. natural implementation doesn't run in constant time: possible timing attacks (especially for PAKE);
  2. constant time implementations are very inefficient, $O(n^4)$;
  3. security is difficult to analyze.

Outline

Shallue-Woestijne-Ulas

First deterministic point construction algorithm on ordinary elliptic curves due to Shallue and Woestijne (ANTS 2006). Later generalized and simplified by Ulas (2007).

Based on Skałba's identity: if $g(x) = x^3 + ax + b$, there are rational functions $X_i(t)$ such that

$$g(X_1(t)) \cdot g(X_2(t)) \cdot g(X_3(t)) = X_4(t)^2$$

Hence, on a finite field, at least one of $g(X_1(t)), g(X_2(t)), g(X_3(t))$ is a square.

Gives a deterministic point construction algorithm, which is efficient if $q \equiv 3 \pmod 4$. Considered for implementation in European e-passports.

Shallue-Woestijne-Ulas

First deterministic point construction algorithm on ordinary elliptic curves due to Shallue and Woestijne (ANTS 2006). Later generalized and simplified by Ulas (2007).

Based on Skałba's identity: if $g(x) = x^3 + ax + b$, there are rational functions $X_i(t)$ such that

$$g(X_1(t)) \cdot g(X_2(t)) \cdot g(X_3(t)) = X_4(t)^2$$

Hence, on a finite field, at least one of $g(X_1(t)), g(X_2(t)), g(X_3(t))$ is a square.

Gives a deterministic point construction algorithm, which is efficient if $q \equiv 3 \pmod 4$. Considered for implementation in European e-passports.

Shallue-Woestijne-Ulas

First deterministic point construction algorithm on ordinary elliptic curves due to Shallue and Woestijne (ANTS 2006). Later generalized and simplified by Ulas (2007).

Based on Skałba's identity: if $g(x) = x^3 + ax + b$, there are rational functions $X_i(t)$ such that

$$g(X_1(t)) \cdot g(X_2(t)) \cdot g(X_3(t)) = X_4(t)^2$$

Hence, on a finite field, at least one of $g(X_1(t)), g(X_2(t)), g(X_3(t))$ is a square.

Gives a deterministic point construction algorithm, which is efficient if $q \equiv 3 \pmod 4$. Considered for implementation in European e-passports.

Icart

Particularly simple deterministic encoding on ordinary elliptic curves when $q \equiv 2 \pmod 3$, presented by Icart at CRYPTO last year. Generalization of the supersingular case.

Defined as $f : u \mapsto (x, y)$ with

$$x = \left(v^2 - b - \frac{u^6}{27}\right)^{1/3} + \frac{u^2}{3} \qquad y = ux + v \qquad v = \frac{3a - u^4}{6u}$$

This simple idea sparked new research into the subject of deterministic hashing into elliptic curves.

Icart

Particularly simple deterministic encoding on ordinary elliptic curves when $q \equiv 2 \pmod 3$, presented by Icart at CRYPTO last year. Generalization of the supersingular case.

Defined as $f\colon u \mapsto (x, y)$ with

$$x = \left(v^2 - b - \frac{u^6}{27}\right)^{1/3} + \frac{u^2}{3} \qquad y = ux + v \qquad v = \frac{3a - u^4}{6u}$$

This simple idea sparked new research into the subject of deterministic hashing into elliptic curves.

Icart

Particularly simple deterministic encoding on ordinary elliptic curves when $q \equiv 2 \pmod 3$, presented by Icart at CRYPTO last year. Generalization of the supersingular case.

Defined as $f : u \mapsto (x, y)$ with

$$x = \left(v^2 - b - \frac{u^6}{27}\right)^{1/3} + \frac{u^2}{3} \qquad y = ux + v \qquad v = \frac{3a - u^4}{6u}$$

This simple idea sparked new research into the subject of deterministic hashing into elliptic curves.

Outline

Statement

In his CRYPTO paper, Icart observed that his function did not reach all points of the curve, and formulated the following conjecture regarding the size of the image.

Conjecture (Icart)

E ordinary elliptic curve over $\mathbb{F}_q$, with $q \equiv 2 \pmod 3$, and $f : \mathbb{F}_q \to E(\mathbb{F}_q)$ Icart's deterministic encoding. There exists a universal constant C such that:

$$\left| \#f(\mathbb{F}_q) - \frac{5}{8} \#E(\mathbb{F}_q) \right| \leq C\sqrt{q}$$

This conjecture, and its generalization to even characteristic as well as to the SWU encoding, is the object of this paper.

Why does Icart's conjecture matter?

- For the encoding to be interesting, its image needs to be large enough.
- Knowing the precise constant makes it possible to assess the security of hash functions based on it.
- Since $5/8 < 1$, Icart's function is neither injective nor surjective.
- However, as $5/8 > 1/2$, a pigeonhole argument ensures that $(u, v) \mapsto f(u) + f(v)$ is surjective. This encoding is quite interesting.
- Most importantly, Icart's conjecture is a nice mathematical problem, and the solution involves interesting results and arguments.

Why does Icart's conjecture matter?

- For the encoding to be interesting, its image needs to be large enough.
- Knowing the precise constant makes it possible to assess the security of hash functions based on it.
- Since $5/8 < 1$, Icart's function is neither injective nor surjective.
- However, as $5/8 > 1/2$, a pigeonhole argument ensures that $(u, v) \mapsto f(u) + f(v)$ is surjective. This encoding is quite interesting.
- Most importantly, Icart's conjecture is a nice mathematical problem, and the solution involves interesting results and arguments.

Why does Icart's conjecture matter?

- For the encoding to be interesting, its image needs to be large enough.
- Knowing the precise constant makes it possible to assess the security of hash functions based on it.
- Since $5/8 < 1$, Icart's function is neither injective nor surjective.
- However, as $5/8 > 1/2$, a pigeonhole argument ensures that $(u, v) \mapsto f(u) + f(v)$ is surjective. This encoding is quite interesting.
- Most importantly, Icart's conjecture is a nice mathematical problem, and the solution involves interesting results and arguments.

Why does Icart's conjecture matter?

- For the encoding to be interesting, its image needs to be large enough.
- Knowing the precise constant makes it possible to assess the security of hash functions based on it.
- Since $5/8 < 1$, Icart's function is neither injective nor surjective.
- However, as $5/8 > 1/2$, a pigeonhole argument ensures that $(u, v) \mapsto f(u) + f(v)$ is surjective. This encoding is quite interesting.
- Most importantly, Icart's conjecture is a nice mathematical problem, and the solution involves interesting results and arguments.

Why does Icart's conjecture matter?

- For the encoding to be interesting, its image needs to be large enough.
- Knowing the precise constant makes it possible to assess the security of hash functions based on it.
- Since $5/8 < 1$, Icart's function is neither injective nor surjective.
- However, as $5/8 > 1/2$, a pigeonhole argument ensures that $(u, v) \mapsto f(u) + f(v)$ is surjective. This encoding is quite interesting.
- Most importantly, Icart's conjecture is a nice mathematical problem, and the solution involves interesting results and arguments.

Outline

Proof sketch

- Key fact: u maps to (x, y) under f if and only if:

$$P(u) = u^4 - 6xu^2 + 6yu - 3a = 0$$

- Regard P as a polynomial over the function field $\mathbb{F}_q(x, y)$ of E. A point (x, y) is in the image if P reduces into a polynomial with a linear factor at place (x, y).
- Assume P is irreducible. The reduction type of P at a given place is related to the cycle decomposition of a certain permutation in its Galois group. There is a linear factor if that permutation has a fixed point.
- A famous theorem by Chebotarev says that the "density" of places with a certain reduction type is equal to the proportion of elements in the Galois group with the corresponding cycle decomposition. Use an effective version to get practical bounds for places of degree 1.
- Then, showing that P is irreducible and computing its Galois group is enough to conclude.

Proof sketch

- Key fact: u maps to (x, y) under f if and only if:

$$P(u) = u^4 - 6xu^2 + 6yu - 3a = 0$$

- Regard P as a polynomial over the function field $\mathbb{F}_q(x, y)$ of E. A point (x, y) is in the image if P reduces into a polynomial with a linear factor at place (x, y).
- Assume P is irreducible. The reduction type of P at a given place is related to the cycle decomposition of a certain permutation in its Galois group. There is a linear factor if that permutation has a fixed point.
- A famous theorem by Chebotarev says that the "density" of places with a certain reduction type is equal to the proportion of elements in the Galois group with the corresponding cycle decomposition. Use an effective version to get practical bounds for places of degree 1.
- Then, showing that P is irreducible and computing its Galois group is enough to conclude.

Proof sketch

- Key fact: u maps to (x, y) under f if and only if:

$$P(u) = u^4 - 6xu^2 + 6yu - 3a = 0$$

- Regard P as a polynomial over the function field $\mathbb{F}_q(x, y)$ of E. A point (x, y) is in the image if P reduces into a polynomial with a linear factor at place (x, y).
- Assume P is irreducible. The reduction type of P at a given place is related to the cycle decomposition of a certain permutation in its Galois group. There is a linear factor if that permutation has a fixed point.
- A famous theorem by Chebotarev says that the "density" of places with a certain reduction type is equal to the proportion of elements in the Galois group with the corresponding cycle decomposition. Use an effective version to get practical bounds for places of degree 1.
- Then, showing that P is irreducible and computing its Galois group is enough to conclude.

Proof sketch

- Key fact: u maps to (x, y) under f if and only if:

$$P(u) = u^4 - 6xu^2 + 6yu - 3a = 0$$

- Regard P as a polynomial over the function field $\mathbb{F}_q(x, y)$ of E. A point (x, y) is in the image if P reduces into a polynomial with a linear factor at place (x, y).
- Assume P is irreducible. The reduction type of P at a given place is related to the cycle decomposition of a certain permutation in its Galois group. There is a linear factor if that permutation has a fixed point.
- A famous theorem by Chebotarev says that the "density" of places with a certain reduction type is equal to the proportion of elements in the Galois group with the corresponding cycle decomposition. Use an effective version to get practical bounds for places of degree 1.
- Then, showing that P is irreducible and computing its Galois group is enough to conclude.

Proof sketch

- Key fact: u maps to (x, y) under f if and only if:

$$P(u) = u^4 - 6xu^2 + 6yu - 3a = 0$$

- Regard P as a polynomial over the function field $\mathbb{F}_q(x, y)$ of E. A point (x, y) is in the image if P reduces into a polynomial with a linear factor at place (x, y).
- Assume P is irreducible. The reduction type of P at a given place is related to the cycle decomposition of a certain permutation in its Galois group. There is a linear factor if that permutation has a fixed point.
- A famous theorem by Chebotarev says that the "density" of places with a certain reduction type is equal to the proportion of elements in the Galois group with the corresponding cycle decomposition. Use an effective version to get practical bounds for places of degree 1.
- Then, showing that P is irreducible and computing its Galois group is enough to conclude.

Outline

Galois groups

P irreducible, separable polynomial of degree n over a field K.

In a suitable extension of F, P has n distinct roots. Let L be the extension of K generated by these roots (splitting field).

Any automorphism of L over K permutes the n roots. The group formed by these permutations is called the Galois group of P. It is a transitive subgroup of S_n.

Example: the Galois group of $u^4 + 1$ over $\mathbb{Q}$ is generated by the double transpositions (12)(34), (13)(24). Indeed, the roots are primitive 8-th roots of unity $\pm\omega, \pm\omega^3$, and the permutations are of the form $\omega \mapsto \pm\omega^k$, $k \in \{1, 3\}$.

Galois groups

P irreducible, separable polynomial of degree n over a field K.

In a suitable extension of F, P has n distinct roots. Let L be the extension of K generated by these roots (splitting field).

Any automorphism of L over K permutes the n roots. The group formed by these permutations is called the Galois group of P. It is a transitive subgroup of S_n.

Example: the Galois group of $u^4 + 1$ over $\mathbb{Q}$ is generated by the double transpositions (12)(34), (13)(24). Indeed, the roots are primitive 8-th roots of unity $\pm\omega, \pm\omega^3$, and the permutations are of the form $\omega \mapsto \pm\omega^k$, $k \in \{1, 3\}$.

Galois groups

P irreducible, separable polynomial of degree n over a field K.

In a suitable extension of F, P has n distinct roots. Let L be the extension of K generated by these roots (splitting field).

Any automorphism of L over K permutes the n roots. The group formed by these permutations is called the Galois group of P. It is a transitive subgroup of S_n.

Example: the Galois group of $u^4 + 1$ over $\mathbb{Q}$ is generated by the double transpositions (12)(34), (13)(24). Indeed, the roots are primitive 8-th roots of unity $\pm\omega, \pm\omega^3$, and the permutations are of the form $\omega \mapsto \pm\omega^k$, $k \in \{1, 3\}$.

Galois groups

P irreducible, separable polynomial of degree n over a field K.

In a suitable extension of F, P has n distinct roots. Let L be the extension of K generated by these roots (splitting field).

Any automorphism of L over K permutes the n roots. The group formed by these permutations is called the Galois group of P. It is a transitive subgroup of S_n.

Example: the Galois group of $u^4 + 1$ over $\mathbb{Q}$ is generated by the double transpositions (12)(34), (13)(24). Indeed, the roots are primitive 8-th roots of unity $\pm\omega, \pm\omega^3$, and the permutations are of the form $\omega \mapsto \pm\omega^k$, $k \in \{1, 3\}$.

Computing Galois groups

Computing Galois groups in general can be difficult.

In our case, we want to compute the Galois group of

$$P(u) = u^4 - 6xu^2 + 6yu - 3a$$

over $\mathbb{F}_q(x, y) = \mathbb{F}_q(x)[y]/(y^2 - x^3 - ax - b)$, so it is made more difficult by 3 parameters that can vary: a, b, q.

However, Galois groups of polynomials of small degree are well-understood. For an irreducible P of degree 4, the Galois group is essentially determined by:

1. whether the discriminant of P is a square;
2. whether the resolvent cubic of P is irreducible.

In our case, we show that P is irreducible, has an irreducible resolvent cubic and a non-square discriminant: its Galois group is S_4.

Computing Galois groups

Computing Galois groups in general can be difficult.

In our case, we want to compute the Galois group of

$$P(u) = u^4 - 6xu^2 + 6yu - 3a$$

over $\mathbb{F}_q(x, y) = \mathbb{F}_q(x)[y]/(y^2 - x^3 - ax - b)$, so it is made more difficult by 3 parameters that can vary: a, b, q.

However, Galois groups of polynomials of small degree are well-understood. For an irreducible P of degree 4, the Galois group is essentially determined by:

1. whether the discriminant of P is a square;
2. whether the resolvent cubic of P is irreducible.

In our case, we show that P is irreducible, has an irreducible resolvent cubic and a non-square discriminant: its Galois group is S_4.

Computing Galois groups

Computing Galois groups in general can be difficult.

In our case, we want to compute the Galois group of

$$P(u) = u^4 - 6xu^2 + 6yu - 3a$$

over $\mathbb{F}_q(x, y) = \mathbb{F}_q(x)[y]/(y^2 - x^3 - ax - b)$, so it is made more difficult by 3 parameters that can vary: a, b, q.

However, Galois groups of polynomials of small degree are well-understood. For an irreducible P of degree 4, the Galois group is essentially determined by:

1. whether the discriminant of P is a square;
2. whether the resolvent cubic of P is irreducible.

In our case, we show that P is irreducible, has an irreducible resolvent cubic and a non-square discriminant: its Galois group is S_4.

Computing Galois groups

Computing Galois groups in general can be difficult.

In our case, we want to compute the Galois group of

$$P(u) = u^4 - 6xu^2 + 6yu - 3a$$

over $\mathbb{F}_q(x, y) = \mathbb{F}_q(x)[y]/(y^2 - x^3 - ax - b)$, so it is made more difficult by 3 parameters that can vary: a, b, q.

However, Galois groups of polynomials of small degree are well-understood. For an irreducible P of degree 4, the Galois group is essentially determined by:

1. whether the discriminant of P is a square;
2. whether the resolvent cubic of P is irreducible.

In our case, we show that P is irreducible, has an irreducible resolvent cubic and a non-square discriminant: its Galois group is S_4.

Computing Galois groups

Computing Galois groups in general can be difficult.

In our case, we want to compute the Galois group of

$$P(u) = u^4 - 6xu^2 + 6yu - 3a$$

over $\mathbb{F}_q(x, y) = \mathbb{F}_q(x)[y]/(y^2 - x^3 - ax - b)$, so it is made more difficult by 3 parameters that can vary: a, b, q.

However, Galois groups of polynomials of small degree are well-understood. For an irreducible P of degree 4, the Galois group is essentially determined by:

1. whether the discriminant of P is a square;
2. whether the resolvent cubic of P is irreducible.

In our case, we show that P is irreducible, has an irreducible resolvent cubic and a non-square discriminant: its Galois group is S_4.

Computing Galois groups

Computing Galois groups in general can be difficult.

In our case, we want to compute the Galois group of

$$P(u) = u^4 - 6xu^2 + 6yu - 3a$$

over $\mathbb{F}_q(x, y) = \mathbb{F}_q(x)[y]/(y^2 - x^3 - ax - b)$, so it is made more difficult by 3 parameters that can vary: a, b, q.

However, Galois groups of polynomials of small degree are well-understood. For an irreducible P of degree 4, the Galois group is essentially determined by:

1. whether the discriminant of P is a square;
2. whether the resolvent cubic of P is irreducible.

In our case, we show that P is irreducible, has an irreducible resolvent cubic and a non-square discriminant: its Galois group is S_4.

Outline

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv (u^2 + u + 2)(u^2 + 2u + 2) \pmod 3$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about 3/4 of the time, and the latter 1/4.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv (u^2 + 2)(u^2 + 3) \pmod 5$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about $3/4$ of the time, and the latter $1/4$.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv (u^2 + 3u + 1)(u^2 + 4u + 1) \pmod 7$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about $3/4$ of the time, and the latter $1/4$.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv (u+2)(u+8)(u+9)(u+15) \pmod{17}$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about $3/4$ of the time, and the latter $1/4$.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv \text{etc.} \pmod{p}$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about $3/4$ of the time, and the latter $1/4$.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv \text{etc.} \pmod{p}$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about $3/4$ of the time, and the latter $1/4$.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv \text{etc.} \pmod{p}$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about $3/4$ of the time, and the latter $1/4$.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Reduction type and Chebotarev

Consider again $Q(u) = u^4 + 1$, and factor it $\bmod p$ for odd primes p:

$$u^4 + 1 \equiv \text{etc.} \pmod{p}$$

It factors either as a product of two irreducible quadratics, or splits completely. Trying many small examples, the former happens about $3/4$ of the time, and the latter $1/4$.

Now recall that the Galois group is $\{(12)(34), (13)(24), (14)(23), (1)(2)(3)(4)\}$.

This is not a coincidence: for each prime $p \neq 2$, there is a corresponding element (or conjugacy class) in the Galois group G, with cycle decomposition equal to the reduction type of Q at p.

The Chebotarev density theorem says that asymptotically, a given reduction type happens for a proportion of primes p equal to $\#C/\#G$, where C is the subset of elements of G with the right cycle decomposition.

Chebotarev for function fields

Similarly, consider $P(u) = u^4 - 6xu^2 + 6yu - 3a$ over $\mathbb{F}_q(x, y)$. We can plug actual points (x, y) of E and factor the resulting polynomial over $\mathbb{F}_q$. The reduction type will correspond to the cycle decomposition of a certain conjugacy class in the Galois group $G = S_4$.

The Chebotarev density theorem still holds: asymptotically, a given reduction type happens for a proportion of "places" equal to $\#C/\#G$ (takes into account points of E over extensions of $\mathbb{F}_q$).

Effective versions of this theorem say that a given reduction types happens for a proportion of $\mathbb{F}_q$-points equal to $\#C/\#G + O(1/\sqrt{q})$.

We are interested in the reduction types $(1, 1, 1, 1)$, $(1, 1, 2)$ and $(1, 3)$ (at least one linear factor). Now S_4 contains 1 permutation of type $(1)(2)(3)(4)$, 6 of type $(1)(2)(34)$ and 8 of type $(1)(234)$, out of a total of 24. Thus the proportion of points on $E(\mathbb{F}_q)$ where P has at least one root is $15/24 + O(1/\sqrt{q})$. QED.

Chebotarev for function fields

Similarly, consider $P(u) = u^4 - 6xu^2 + 6yu - 3a$ over $\mathbb{F}_q(x, y)$. We can plug actual points (x, y) of E and factor the resulting polynomial over $\mathbb{F}_q$. The reduction type will correspond to the cycle decomposition of a certain conjugacy class in the Galois group $G = S_4$.

The Chebotarev density theorem still holds: asymptotically, a given reduction type happens for a proportion of "places" equal to $\#C/\#G$ (takes into account points of E over extensions of $\mathbb{F}_q$).

Effective versions of this theorem say that a given reduction types happens for a proportion of $\mathbb{F}_q$-points equal to $\#C/\#G + O(1/\sqrt{q})$.

We are interested in the reduction types $(1, 1, 1, 1)$, $(1, 1, 2)$ and $(1, 3)$ (at least one linear factor). Now S_4 contains 1 permutation of type $(1)(2)(3)(4)$, 6 of type $(1)(2)(34)$ and 8 of type $(1)(234)$, out of a total of 24. Thus the proportion of points on $E(\mathbb{F}_q)$ where P has at least one root is $15/24 + O(1/\sqrt{q})$. QED.

Chebotarev for function fields

Similarly, consider $P(u) = u^4 - 6xu^2 + 6yu - 3a$ over $\mathbb{F}_q(x, y)$. We can plug actual points (x, y) of E and factor the resulting polynomial over $\mathbb{F}_q$. The reduction type will correspond to the cycle decomposition of a certain conjugacy class in the Galois group $G = S_4$.

The Chebotarev density theorem still holds: asymptotically, a given reduction type happens for a proportion of “places” equal to $\#C/\#G$ (takes into account points of E over extensions of $\mathbb{F}_q$).

Effective versions of this theorem say that a given reduction types happens for a proportion of $\mathbb{F}_q$-points equal to $\#C/\#G + O(1/\sqrt{q})$.

We are interested in the reduction types $(1, 1, 1, 1)$, $(1, 1, 2)$ and $(1, 3)$ (at least one linear factor). Now S_4 contains 1 permutation of type $(1)(2)(3)(4)$, 6 of type $(1)(2)(34)$ and 8 of type $(1)(234)$, out of a total of 24. Thus the proportion of points on $E(\mathbb{F}_q)$ where P has at least one root is $15/24 + O(1/\sqrt{q})$. QED.

Chebotarev for function fields

Similarly, consider $P(u) = u^4 - 6xu^2 + 6yu - 3a$ over $\mathbb{F}_q(x, y)$. We can plug actual points (x, y) of E and factor the resulting polynomial over $\mathbb{F}_q$. The reduction type will correspond to the cycle decomposition of a certain conjugacy class in the Galois group $G = S_4$.

The Chebotarev density theorem still holds: asymptotically, a given reduction type happens for a proportion of "places" equal to $\#C/\#G$ (takes into account points of E over extensions of $\mathbb{F}_q$).

Effective versions of this theorem say that a given reduction types happens for a proportion of $\mathbb{F}_q$-points equal to $\#C/\#G + O(1/\sqrt{q})$.

We are interested in the reduction types $(1, 1, 1, 1)$, $(1, 1, 2)$ and $(1, 3)$ (at least one linear factor). Now S_4 contains 1 permutation of type $(1)(2)(3)(4)$, 6 of type $(1)(2)(34)$ and 8 of type $(1)(234)$, out of a total of 24. Thus the proportion of points on $E(\mathbb{F}_q)$ where P has at least one root is $15/24 + O(1/\sqrt{q})$. QED.

Outline

Even characteristic

There is a variant of Icart's function in characteristic 2, and the proof carries over to this variant with almost no change.

Only subtlety: the computation of quartic Galois groups is different in characteristic 2 (one has to replace the discriminant by a "resolvent quadratic" polynomial to decide whether the group is contained in A_4 or not).

(We actually got this part wrong in the proceedings version. Please check out ePrint Report 2010/037 for a correct proof).

Even characteristic

There is a variant of Icart's function in characteristic 2, and the proof carries over to this variant with almost no change.

Only subtlety: the computation of quartic Galois groups is different in characteristic 2 (one has to replace the discriminant by a "resolvent quadratic" polynomial to decide whether the group is contained in A_4 or not).

(We actually got this part wrong in the proceedings version. Please check out ePrint Report 2010/037 for a correct proof).

Even characteristic

There is a variant of Icart's function in characteristic 2, and the proof carries over to this variant with almost no change.

Only subtlety: the computation of quartic Galois groups is different in characteristic 2 (one has to replace the discriminant by a "resolvent quadratic" polynomial to decide whether the group is contained in A_4 or not).

(We actually got this part wrong in the proceedings version. Please check out ePrint Report 2010/037 for a correct proof).

Shallue-Woestijne-Ulas

We are also able to compute the image size of the simplified SWU encoding introduced by Coron and Icart.

The method is mostly the same, except that the image of the encoding comes in two "pieces" according as which of the two possible values of x is the right one for a given value of the parameter.

The Chebotarev method applies to those pieces. We find that the corresponding Galois group is D_8 for both halves, giving a proportion of points in the image equal $3/8 + O(1/\sqrt{q})$ overall.

Shallue-Woestijne-Ulas

We are also able to compute the image size of the simplified SWU encoding introduced by Coron and Icart.

The method is mostly the same, except that the image of the encoding comes in two "pieces" according as which of the two possible values of x is the right one for a given value of the parameter.

The Chebotarev method applies to those pieces. We find that the corresponding Galois group is D_8 for both halves, giving a proportion of points in the image equal $3/8 + O(1/\sqrt{q})$ overall.

Shallue-Woestijne-Ulas

We are also able to compute the image size of the simplified SWU encoding introduced by Coron and Icart.

The method is mostly the same, except that the image of the encoding comes in two "pieces" according as which of the two possible values of x is the right one for a given value of the parameter.

The Chebotarev method applies to those pieces. We find that the corresponding Galois group is D_8 for both halves, giving a proportion of points in the image equal $3/8 + O(1/\sqrt{q})$ overall.

Summary and Outlook

- Icart's conjecture is true.
- We can prove analogues for characteristic 2, for SWU, and the same method generalizes to all algebraic encodings to curves (many recent examples).
- The proof uses nice algebraic tools (Chebotarev) which the mathematical cryptographer can find of interest in other situations.

Further problems:

- Collision probability.
- Carry out the computations for recently proposed encodings?
- Mechanical method to prove irreducibility and compute Galois groups? (possible in principle, but...).

Summary and Outlook

- Icart's conjecture is true.
- We can prove analogues for characteristic 2, for SWU, and the same method generalizes to all algebraic encodings to curves (many recent examples).
- The proof uses nice algebraic tools (Chebotarev) which the mathematical cryptographer can find of interest in other situations.

Further problems:

- Collision probability.
- Carry out the computations for recently proposed encodings?
- Mechanical method to prove irreducibility and compute Galois groups? (possible in principle, but...).

Summary and Outlook

- Icart's conjecture is true.
- We can prove analogues for characteristic 2, for SWU, and the same method generalizes to all algebraic encodings to curves (many recent examples).
- The proof uses nice algebraic tools (Chebotarev) which the mathematical cryptographer can find of interest in other situations.

Further problems:

- Collision probability.
- Carry out the computations for recently proposed encodings?
- Mechanical method to prove irreducibility and compute Galois groups? (possible in principle, but...).

Summary and Outlook

- Icart's conjecture is true.
- We can prove analogues for characteristic 2, for SWU, and the same method generalizes to all algebraic encodings to curves (many recent examples).
- The proof uses nice algebraic tools (Chebotarev) which the mathematical cryptographer can find of interest in other situations.

Further problems:

- Collision probability.
- Carry out the computations for recently proposed encodings?
- Mechanical method to prove irreducibility and compute Galois groups? (possible in principle, but...).

Summary and Outlook

- Icart's conjecture is true.
- We can prove analogues for characteristic 2, for SWU, and the same method generalizes to all algebraic encodings to curves (many recent examples).
- The proof uses nice algebraic tools (Chebotarev) which the mathematical cryptographer can find of interest in other situations.

Further problems:

- Collision probability.
- Carry out the computations for recently proposed encodings?
- Mechanical method to prove irreducibility and compute Galois groups? (possible in principle, but...).

Summary and Outlook

- Icart's conjecture is true.
- We can prove analogues for characteristic 2, for SWU, and the same method generalizes to all algebraic encodings to curves (many recent examples).
- The proof uses nice algebraic tools (Chebotarev) which the mathematical cryptographer can find of interest in other situations.

Further problems:

- Collision probability.
- Carry out the computations for recently proposed encodings?
- Mechanical method to prove irreducibility and compute Galois groups? (possible in principle, but...).

Thank you!