Elliptic curves in Lean

Mathematical Theorem Proving Workshop

David Kurniadi Angdinata

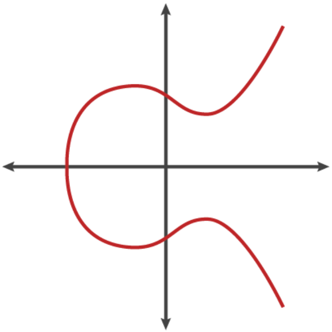
London School of Geometry and Number Theory

Monday, 25 April 2022

Informally

What are elliptic curves?

- Solutions to $y^2 = x^3 + Ax + B$.



- Points form a group!

Motivation

Why do we care?

Public-key cryptography (over a large finite field)

- Integer factorisation (*e.g.* Lenstra's method)
  Breaks the RSA cryptosystem
- Diffie–Hellman key exchange
  Discrete logarithm (solve $nQ = P$ given P and Q)
- Supersingular isogeny Diffie–Hellman key exchange

Number theory (over a field/ring/scheme)

- The simplest non-trivial objects in algebraic geometry
- Rational elliptic curve associated to $a^p + b^p = c^p$ cannot be modular
  But rational elliptic curves are modular (modularity theorem)
- Distribution of ranks of rational elliptic curves
  The BSD conjecture (analytic rank equals algebraic rank)

Globally

An **elliptic curve E over a scheme S** is a diagram

$$
\begin{tikzcd}
E \arrow[d, "f"'] \\
S \arrow[u, bend right, "0"']
\end{tikzcd}
$$

with a few technical conditions. [1]

For a scheme T over S, define the set of **T-points** of E by

$$E(T) := \mathrm{Hom}_S(T, E),$$

which is naturally identified with a *Picard group* $\mathrm{Pic}^0_{E/S}(T)$ of E.

This defines a contravariant functor $\mathbf{Sch}_S \to \mathbf{Ab}$ given by $T \mapsto E(T)$.

Good for algebraic geometry, but not very friendly...

[1] f is smooth, proper, and all its geometric fibres are integral curves of genus one

Locally

Let T/S be a field extension K/F. [2]

An **elliptic curve** E **over a field** F is a tuple $(E, 0)$.

- E is a nice [3] genus one curve over F.
- 0 is an F-point.

The Picard group becomes

$$\mathrm{Pic}^0_{E/F}(K) = \frac{\{\text{degree zero divisors of } E \text{ over } K\}}{\{\text{principal divisors of } E \text{ over } K\}}.$$

This defines a covariant functor $\mathbf{Alg}_F \to \mathbf{Ab}$ given by $K \mapsto E(K)$.

Group law is free, but still need equations...

[2] $S = \mathrm{Spec}(F)$ and $T = \mathrm{Spec}(K)$, or even rings whose class group has no 12-torsion
[3] smooth, proper, and geometrically integral

Concretely

The Riemann–Roch theorem gives *Weierstrass equations*.

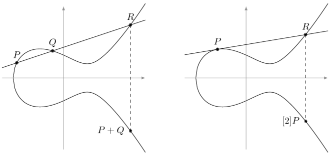
$E(K)$ is basically the set of solutions $(x, y) \in K^2$ to

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \qquad a_i \in F.$$

If $\mathrm{char}(F) \neq 2, 3$, can reduce this to

$$y^2 = x^3 + Ax + B, \qquad A, B \in F.$$

The *group law* is reduced to drawing lines.



Implementation

Three definitions of elliptic curves:

1. Abstract definition over a scheme
2. Abstract definition over a field
3. Concrete definition over a field

Generality: $1 \supset 2 \stackrel{\text{RR}}{=} 3$

- 1. and 2. require much algebraic geometry (properness, genus, ...)
- 2. $=$ 3. also requires algebraic geometry (divisors, differentials, ...)
- 3. requires just five coefficients (and a non-zero *discriminant*)!

```
def disc_aux {R : Type} [comm_ring R] (a₁ a₂ a₃ a₄ a₆ : R) : R :=
 -(a₁^2 + 4*a₂)^2*(a₁^2*a₆ + 4*a₂*a₆ - a₁*a₃*a₄ + a₂*a₃^2 - a₄^2)
 - 8*(2*a₄ + a₁*a₃)^3 - 27*(a₃^2 + 4*a₆)^2
 + 9*(a₁^2 + 4*a₂)*(2*a₄ + a₁*a₃)*(a₃^2 + 4*a₆)

structure EllipticCurve (R : Type) [comm_ring R] :=
 (a₁ a₂ a₃ a₄ a₆ : R) (disc : units R) (disc_eq : disc.val = disc_aux a₁ a₂ a₃ a₄ a₆)
```

This is the *scheme* E, but what about the *abelian group* $E(F)$?

Points

```
variables {F : Type} [field F] (E : EllipticCurve F) (K : Type) [field K] [algebra F K]

inductive point
  | zero
  | some (x y : K) (w : y^2 + E.a₁*x*y + E.a₃*y = x^3 + E.a₂*x^2 + E.a₄*x + E.a₆)

notation E(K) := point E K
```

Identity

```
instance : has_zero E(K) := ⟨zero⟩
```

Negation

```
def neg : E(K) → E(K)
  | zero := zero
  | (some x y w) := some x (-y - E.a₁*x - E.a₃) $ by { rw [← w], ring }

instance : has_neg E(K) := ⟨neg⟩
```

Points

```
variables {F : Type} [field F] (E : EllipticCurve F) (K : Type) [field K] [algebra F K]

inductive point
  | zero
  | some (x y : K) (w : y^2 + E.a₁*x*y + E.a₃*y = x^3 + E.a₂*x^2 + E.a₄*x + E.a₆)

notation E(K) := point E K
```

Addition

```
def add : E(K) → E(K) → E(K)
  | zero P := P
  | P zero := P
  | (some x₁ y₁ w₁) (some x₂ y₂ w₂) :=
    if x_ne : x₁ ≠ x₂ then
     let L := (y₁ - y₂) / (x₁ - x₂),
         x₃ := L^2 + E.a₁*L - E.a₂ - x₁ - x₂,
         y₃ := -L*x₃ - E.a₁*x₃ - y₁ + L*x₁ - E.a₃
     in some x₃ y₃ $ by { ... }
    else if y_ne : y₁ + y₂ + E.a₁*x₂ + E.a₃ ≠ 0 then
     ...
    else
     zero

instance : has_add E(K) := ⟨add⟩
```

Points

```
variables {F : Type} [field F] (E : EllipticCurve F) (K : Type) [field K] [algebra F K]

inductive point
  | zero
  | some (x y : K) (w : y^2 + E.a₁*x*y + E.a₃*y = x^3 + E.a₂*x^2 + E.a₄*x + E.a₆)

notation E(K) := point E K
```

Commutativity is doable

```
lemma add_comm (P Q : E(K)) : P + Q = Q + P :=
  by { rcases ⟨P, Q⟩ with ⟨_ | _, _ | _⟩, . . . }
```

Associativity is difficult

```
lemma add_assoc (P Q R : E(K)) : (P + Q) + R = P + (Q + R) :=
  by { rcases ⟨P, Q, R⟩ with ⟨_ | _, _ | _, _ | _⟩, . . . }
```

Associativity

Known to be very difficult with several proofs:

- Just do it!
  (times out with 130,000(!?) coefficients)
- Uniformisation
  (requires complex analysis and modular forms)
- Cayley–Bacharach
  (requires incidence geometry notions and Bézout's theorem)
- $E(K) \cong \mathrm{Pic}^0_{E/F}(K)$
  (requires divisors, differentials, and the Riemann–Roch theorem)

Current status:

- Left as a `sorry`
- Attempt (by M Masdeu) to bash it out using `linear_combination`
- Proved (by E-I Bartzia and P-Y Strub) in Coq [4]
  that $E(K) \cong \mathrm{Pic}^0_{E/F}(K)$ for $\mathrm{char}(F) \neq 2, 3$

[4] A Formal Library for Elliptic Curves in the Coq Proof Assistant (2015)

Progress

Modulo associativity, what has been done?

Functoriality $\mathbf{Alg}_F \to \mathbf{Ab}$

```
def point_hom (φ : K →ₐ[F] L) : E(K) → E(L)
  | zero := zero
  | (some x y w) := some (φ x) (φ y) $ by { ... }

local notation K →[F] L := (algebra.of_id K L).restrict_scalars F

lemma point_hom.id (P : E(K)) : point_hom (K →[F] K) P = P := by cases P; refl

lemma point_hom.comp (P : E(K)) :
  point_hom (L →[F] M) (point_hom (K →[F] L) P) =
    point_hom ((L →[F] M).comp (K →[F] L)) P := by cases P; refl
```

Galois module structure $\mathrm{Gal}(L/K) \curvearrowright E(L)$

```
def point_gal (σ : L ≃ₐ[K] L) : E(L) → E(L)
  | zero := zero
  | (some x y w) := some (σ • x) (σ • y) $ by { ... }

lemma point_gal.fixed :
  mul_action.fixed_points (L ≃ₐ[K] L) E(L) = (point_hom (K →[F] L)).range := by { ... }
```

Progress

Modulo associativity, what has been done?

Isomorphisms $(x, y) \mapsto (u^2x + r, u^3y + u^2sx + t)$

```
variables (u : units F) (r s t : F)

def cov : EllipticCurve F :=
{ a₁ := u.inv*(E.a₁ + 2*s),
  a₂ := u.inv^2*(E.a₂ − s*E.a₁ + 3*r − s^2),
  a₃ := u.inv^3*(E.a₃ + r*E.a₁ + 2*t),
  a₄ := u.inv^4*(E.a₄ − s*E.a₃ + 2*r*E.a₂ − (t + r*s)*E.a₁ + 3*r^2 − 2*s*t),
  a₆ := u.inv^6*(E.a₆ + r*E.a₄ + r^2*E.a₂ + r^3 − t*E.a₃ − t^2 − r*t*E.a₁),
  disc := ⟨u.inv^12*E.disc.val, u.val^12*E.disc.inv, by { ... }, by { ... }⟩,
  disc_eq := by { simp only, rw [disc_eq, disc_aux, disc_aux], ring } }

def cov.to_fun : (E.cov u r s t)(K) → E(K)
  | zero := zero
  | (some x y w) := some (u.val^2*x + r) (u.val^3*y + u.val^2*s*x + t) $ by { ... }

def cov.inv_fun : E(K) → (E.cov u r s t)(K)
  | zero := zero
  | (some x y w) := some (u.inv^2*(x − r)) (u.inv^3*(y − s*x + r*s − t)) $ by { ... }

def cov.equiv_add : (E.cov u r s t)(K) ≃+ E(K) :=
  ⟨cov.to_fun u r s t, cov.inv_fun u r s t, by { ... }, by { ... }, by { ... }⟩
```

Progress

Modulo associativity, what has been done?

2-division polynomial $\psi_2(x)$

```
def ψ₂_x : cubic K := ⟨4, E.a₁^2 + 4*E.a₂, 4*E.a₄ + 2*E.a₁*E.a₃, E.a₃^2 + 4*E.a₆⟩

lemma ψ₂_x.disc_eq_disc : (ψ₂_x E K).disc = 16*E.disc := by { . . . }
```

Structure of $E(K)[2]$

```
notation E(K)[n] := ((·) n : E(K) →+ E(K)).ker

lemma E₂.x {x y w} : some x y w ∈ E(K)[2] ↔ x ∈ (ψ₂_x E K).roots := by { . . . }

theorem E₂.card_le_four : fintype.card E(K)[2] ≤ 4 := by { . . . }

variables [algebra ((ψ₂_x E F).splitting_field) K]

theorem E₂.card_eq_four : fintype.card E(K)[2] = 4 := by { . . . }

lemma E₂.gal_fixed (σ : L ≃ₐ[K] L) (P : E(L)[2]) : σ · P = P := by { . . . }
```

Mordell–Weil

Let K be a number field. Then $E(K)$ is finitely generated.

Show $E(K)/2E(K)$ is finite:

1. Reduce to $E[2] \subset E(K)$
2. Define 2-descent $\delta : E(K)/2E(K) \hookrightarrow K^\times/(K^\times)^2 \times K^\times/(K^\times)^2$
3. Show that $\operatorname{im}\delta \leq K(\emptyset, 2) \times K(\emptyset, 2)$
4. Prove exactness of $0 \to \mathcal{O}_K^\times/(\mathcal{O}_K^\times)^n \to K(\emptyset, n) \to \mathrm{Cl}_K[n] \to 0$
5. Apply finiteness of $\mathcal{O}_K^\times/(\mathcal{O}_K^\times)^n$ and $\mathrm{Cl}_K[n]$

Show this implies $E(K)$ is finitely generated:

1. Define heights on elliptic curves
2. (J Zhang) Prove the descent theorem

Soon: Mordell's theorem for $E[2] \subset E(\mathbb{Q})$.

Future

Potential future projects:

- n-division polynomials and the structure of $E(K)[n]$
- formal groups and local theory
- ramification theory $\implies$ Mordell–Weil theorem for number fields
- Galois cohomology $\implies$ Selmer and Tate–Shafarevich groups
- modular forms $\implies$ complex theory
- algebraic geometry $\implies$ proof of associativity