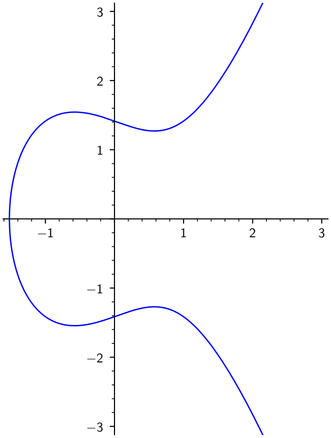
유한체에서의 타원곡선과 암호학에서의 응용

22학번 손량

May 21, 2022

타원곡선

Definition



체 K에 대해, $\mathrm{char}(K) \notin \{2,3\}$이면 K 위에서 정의된 타원 곡선은 다음 방정식의 해집합.

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

여기서 $\Delta = -16(4a^3 + 27b^2) \neq 0$이어야 함.

유한체

Definition

덧셈과 곱셈 두 개의 이항 연산을 가지는 집합 F

유한체

Definition

덧셈과 곱셈 두 개의 이항 연산을 가지는 집합 F

- 이항 연산은 $F \times F \to F$여야 함

유한체

Definition

덧셈과 곱셈 두 개의 이항 연산을 가지는 집합 F

- 이항 연산은 $F \times F \to F$여야 함
- $a + (b + c) = (a + b) + c,\ a \cdot (b \cdot c) = (a \cdot b) \cdot c$ (결합법칙)
- $a + b = b + a,\ a \cdot b = b \cdot a$ (교환법칙)
- $a + 0 = a,\ a \cdot 1 = a$인 $0, 1 \in F$ 존재 (항등원)
- $a + (-a) = 0$인 $-a \in F$ 존재 (덧셈의 역원)
- $a \neq 0$인 모든 a에 대해, $a \cdot a^{-1} = 1$인 $a^{-1} \in F$ 존재 (곱셈의 역원)
- $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ (분배법칙)

유한체

Definition

덧셈과 곱셈 두 개의 이항 연산을 가지는 집합 F

- 이항 연산은 $F \times F \to F$여야 함
- $a + (b + c) = (a + b) + c,\ a \cdot (b \cdot c) = (a \cdot b) \cdot c$ (결합법칙)
- $a + b = b + a,\ a \cdot b = b \cdot a$ (교환법칙)
- $a + 0 = a,\ a \cdot 1 = a$인 $0, 1 \in F$ 존재 (항등원)
- $a + (-a) = 0$인 $-a \in F$ 존재 (덧셈의 역원)
- $a \neq 0$인 모든 a에 대해, $a \cdot a^{-1} = 1$인 $a^{-1} \in F$ 존재 (곱셈의 역원)
- $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ (분배법칙)

대표적인 체의 예: 유리수 $\mathbb{Q}$, 실수 $\mathbb{R}$

유한체

Definition of Field

유한체: order(원소의 개수)가 유한한 체

- 유한체의 order는 $q = p^k$ 형태를 가짐
- 유한체의 예: $\mathbb{F}_p, \mathbb{F}_{2^m}$ 등

유한체

유한체 $\mathbb{F}_p$

- $\mathbb{F}_p$는 원소를 나열해 $\{0, 1, \cdots, p-1\}$으로 나타낼 수 있음
- 덧셈과 곱셈은 직접 곱하거나 더한 결과의 p로 나눈 나머지를 취한 것으로 정의.

유한체

유한체 $\mathbb{F}_p$

- $\mathbb{F}_p$는 원소를 나열해 $\{0, 1, \cdots, p-1\}$으로 나타낼 수 있음
- 덧셈과 곱셈은 직접 곱하거나 더한 결과의 p로 나눈 나머지를 취한 것으로 정의.
- 예를 들어, $\mathbb{F}_2$에서 다음이 성립.
  - $1+1=0$
  - $(x+y)^2 = x^2 + y^2$ ('1학년의 꿈')

유한체

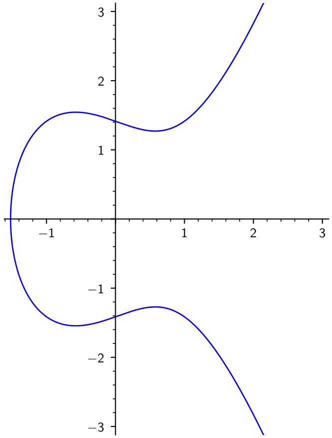
유한체 $\mathbb{F}_p$

- $\mathbb{F}_p$는 원소를 나열해 $\{0, 1, \cdots, p-1\}$으로 나타낼 수 있음
- 덧셈과 곱셈은 직접 곱하거나 더한 결과의 p로 나눈 나머지를 취한 것으로 정의.
- 예를 들어, $\mathbb{F}_2$에서 다음이 성립.
  - $1 + 1 = 0$
  - $(x+y)^2 = x^2 + y^2$ ('1학년의 꿈')
- 덧셈의 역원: $a + x \equiv 0 \pmod p$의 유일한 해로 정의
- 곱셈의 역원: $a \cdot x \equiv 1 \pmod p$의 유일한 해로 정의

- $\mathbb{F}_p$는 원소를 나열해 $\{0, 1, \cdots, p-1\}$으로 나타낼 수 있음
- 덧셈과 곱셈은 직접 곱하거나 더한 결과의 p로 나눈 나머지를 취한 것으로 정의.
- 예를 들어, $\mathbb{F}_2$에서 다음이 성립.
  - $1 + 1 = 0$
  - $(x + y)^2 = x^2 + y^2$ ('1학년의 꿈')
- 덧셈의 역원: $a + x \equiv 0 \pmod p$의 유일한 해로 정의
- 곱셈의 역원: $a \cdot x \equiv 1 \pmod p$의 유일한 해로 정의
- 이 $\mathbb{F}_p$에서 타원곡선을 정의할 수 있음
  - 타원곡선의 형태를 결정하는 매개변수 a, b가 $\mathbb{F}_p$에 존재.

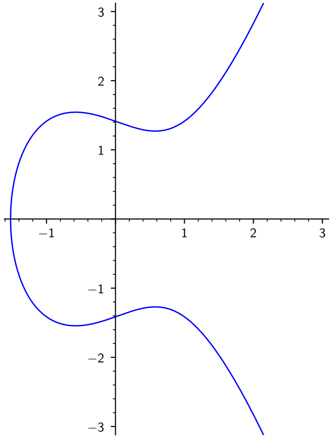
유한체

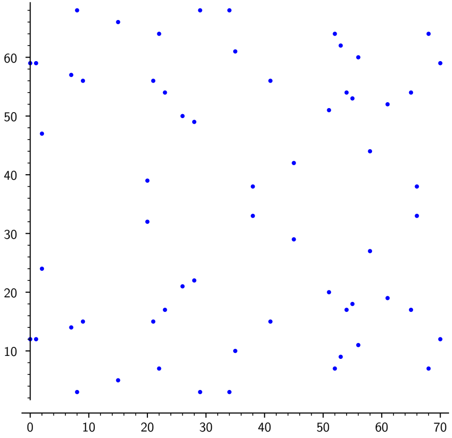
$\mathbb{F}_{71}$ 위에서 정의된 타원곡선 $y^2 = x^3 - x + 2$



유한체

$\mathbb{F}_{71}$ 위에서 정의된 타원곡선 $y^2 = x^3 - x + 2$





유한체

유한체 $\mathbb{F}_{2^m}$

나중에 시간이 나면 다룰 내용

타원곡선에서 정의된 군

$\mathbb{F}_p$의 경우: $E(\mathbb{F}_p)$

Point at infinity: $\mathcal{O}$

- 자기 자신 더하기: $\mathcal{O} + \mathcal{O} = \mathcal{O}$
- 다른 점 더하기: $(x, y) + \mathcal{O} = \mathcal{O} + (x, y) = (x, y) \quad \forall (x, y) \in E(\mathbb{F}_p)$

타원곡선에서 정의된 군

$\mathbb{F}_p$의 경우: $E(\mathbb{F}_p)$

Point at infinity: $\mathcal{O}$

- 자기 자신 더하기: $\mathcal{O} + \mathcal{O} = \mathcal{O}$
- 다른 점 더하기: $(x, y) + \mathcal{O} = \mathcal{O} + (x, y) = (x, y) \quad \forall (x, y) \in E(\mathbb{F}_p)$

두 점의 덧셈

- 덧셈 역원: $(x, y) + (x, -y) = \mathcal{O} \quad \forall (x, y) \in E(\mathbb{F}_p)$
- $(x_3, y_3) := (x_1, y_1) + (x_2, y_2) \quad (x_1 \neq x_2)$

타원곡선에서 정의된 군

$\mathbb{F}_p$ 의 경우: $E(\mathbb{F}_p)$

Point at infinity: $\mathcal{O}$

- 자기 자신 더하기: $\mathcal{O} + \mathcal{O} = \mathcal{O}$
- 다른 점 더하기: $(x, y) + \mathcal{O} = \mathcal{O} + (x, y) = (x, y) \quad \forall (x, y) \in E(\mathbb{F}_p)$

두 점의 덧셈

- 덧셈 역원: $(x, y) + (x, -y) = \mathcal{O} \quad \forall (x, y) \in E(\mathbb{F}_p)$
- $(x_3, y_3) := (x_1, y_1) + (x_2, y_2) \quad (x_1 \neq x_2)$
  - $\lambda := \frac{y_2 - y_1}{x_2 - x_1}$
  - $x_3 := \lambda^2 - x_1 - x_2$
  - $y_3 := \lambda(x_1 - x_3) - y_1$

타원곡선에서 정의된 군

$\mathbb{F}_p$의 경우: $E(\mathbb{F}_p)$

두 점의 덧셈

- $(x_3, y_3) := (x_1, y_1) + (x_1, y_1)$

타원곡선에서 정의된 군

$\mathbb{F}_p$ 의 경우: $E(\mathbb{F}_p)$

두 점의 덧셈

- $(x_3, y_3) := (x_1, y_1) + (x_1, y_1)$
  - $\lambda := \frac{3x_1^2 + a}{2y_1}$
  - $x_3 := \lambda^2 - 2x_1$
  - $y_3 := \lambda(x_1 - x_3) - y_1$

두 점의 덧셈

- $(x_3, y_3) := (x_1, y_1) + (x_1, y_1)$
  - $\lambda := \frac{3x_1^2 + a}{2y_1}$
  - $x_3 := \lambda^2 - 2x_1$
  - $y_3 := \lambda(x_1 - x_3) - y_1$
- $n(x, y) = \underbrace{(x, y) + (x, y) + \cdots + (x, y)}_{n\text{개 더함}}$

타원곡선에서 정의된 군

$\mathbb{F}_{2^m}$ 의 경우

나중에 시간이 나면 다룰 내용

타원곡선에서 정의된 군

Double-and-Add Algorithm

Elliptic Curve Cryptography에서 n회 덧셈은 많이 쓰이는데, 여기서 n은 엄청 클 수 있음.

타원곡선에서 정의된 군

Double-and-Add Algorithm

Elliptic Curve Cryptography에서 n회 덧셈은 많이 쓰이는데, 여기서 n은 엄청 클 수 있음. $O(n)$인 naive한 알고리즘은 너무 느리고, 실제 구현에서는 $O(\log_2 n)$ 알고리즘을 사용.

1. $n \in \mathbb{F}_p$, $P \in E(\mathbb{F}_p)$에 대해, nP를 Q에 저장한다고 하자. 초기에 $Q = \mathcal{O}$.
2. n의 most significant bit부터 시작, least significant bit까지 반복.

타원곡선에서 정의된 군

Double-and-Add Algorithm

Elliptic Curve Cryptography에서 n회 덧셈은 많이 쓰이는데, 여기서 n은 엄청 클 수 있음. $O(n)$인 naive한 알고리즘은 너무 느리고, 실제 구현에서는 $O(\log_2 n)$ 알고리즘을 사용.

1. $n \in \mathbb{F}_p$, $P \in E(\mathbb{F}_p)$에 대해, nP를 Q에 저장한다고 하자. 초기에 $Q = \mathcal{O}$.
2. n의 most significant bit부터 시작, least significant bit까지 반복.
   2.1 $2Q \to Q$
   2.2 지금 보고 있는 비트를 `b`라고 할 때, `b == 1`이면 $Q + P \to Q$

타원곡선에서 정의된 군

Double-and-Add Algorithm

Elliptic Curve Cryptography에서 n회 덧셈은 많이 쓰이는데, 여기서 n은 엄청 클 수 있음. $O(n)$인 naive한 알고리즘은 너무 느리고, 실제 구현에서는 $O(\log_2 n)$ 알고리즘을 사용.

1. $n \in \mathbb{F}_p$, $P \in E(\mathbb{F}_p)$에 대해, nP를 Q에 저장한다고 하자. 초기에 $Q = \mathcal{O}$.
2. n의 most significant bit부터 시작, least significant bit까지 반복.
   2.1 $2Q \to Q$
   2.2 지금 보고 있는 비트를 `b`라고 할 때, `b == 1`이면 $Q + P \to Q$

이 알고리즘을 실제 코드로 구현할 때에는 side-channel attack을 막기 위한 처리가 필요

타원곡선의 응용

Elliptic Curve Domain Parameters

암호화, 복호화를 하기 위한 타원곡선, 유한체의 조건을 미리 정해놓아야 함

$$T = (p, a, b, G, n, h)$$

- p: 유한체 $\mathbb{F}_p$의 order
- a, b: 타원곡선 방정식 $y^2 = x^3 + ax + b$의 파라메터
- G: 타원곡선 위의 base point
- n: G의 order
- h: cofactor $\#E(\mathbb{F}_p)/n$

key pair는 무작위로 고른 정수 $d \in [1, n-1]$에 대해 $Q = dG$를 계산, (d, Q).

- 여기서 d가 비밀키, Q가 공개키.

key pair는 무작위로 고른 정수 $d \in [1, n-1]$에 대해 $Q = dG$를 계산, (d, Q).

- 여기서 d가 비밀키, Q가 공개키.

Alice, Bob이 각각 key pair $(d_A, Q_A), (d_B, Q_B)$를 갖고 있다고 하자. 두 사람이 공통의 기밀 정보 z를 공유하고 싶다고 하자.

1. Alice는 Bob에게 Q_A를, Bob은 Alice에게 Q_B를 보냄
2. Alice는 $z = d_A Q_B$, Bob은 $z = d_B Q_A$를 계산

key pair는 무작위로 고른 정수 $d \in [1, n-1]$에 대해 $Q = dG$를 계산, (d, Q).

- 여기서 d가 비밀키, Q가 공개키.

Alice, Bob이 각각 key pair $(d_A, Q_A), (d_B, Q_B)$를 갖고 있다고 하자. 두 사람이 공통의 기밀 정보 z를 공유하고 싶다고 하자.

1. Alice는 Bob에게 Q_A를, Bob은 Alice에게 Q_B를 보냄
2. Alice는 $z = d_A Q_B$, Bob은 $z = d_B Q_A$를 계산

$z = d_A Q_B = d_B Q_A = d_A d_B G$이므로, Alice와 Bob은 공통의 기밀 정보를 가지게 된다.

이산 로그

Alice, Bob의 통신 과정을 Eve가 도청했다고 가정

이산 로그

Computational Complexity

Alice, Bob의 통신 과정을 Eve가 도청했다고 가정

- Eve가 얻은 Q_A, Q_B를 이용하여 z를 계산하기 위해서는 d_A 혹은 d_B를 계산해야 함
- 이는 이산 로그 문제로, 군의 order n에 대해 적어도 $O(\sqrt{n})$의 시간 복잡도를 가짐

이산 로그

Computational Complexity

Alice, Bob의 통신 과정을 Eve가 도청했다고 가정

- Eve가 얻은 Q_A, Q_B를 이용하여 z를 계산하기 위해서는 d_A 혹은 d_B를 계산해야 함
- 이는 이산 로그 문제로, 군의 order n에 대해 적어도 $O(\sqrt{n})$의 시간 복잡도를 가짐
  - $O(\sqrt{n})$보다 효율적인 알고리즘이 있는지는 미해결 문제

이산 로그

Alice, Bob의 통신 과정을 Eve가 도청했다고 가정

- Eve가 얻은 Q_A, Q_B를 이용하여 z를 계산하기 위해서는 d_A 혹은 d_B를 계산해야 함
- 이는 이산 로그 문제로, 군의 order n에 대해 적어도 $O(\sqrt{n})$의 시간 복잡도를 가짐
  - $O(\sqrt{n})$보다 효율적인 알고리즘이 있는지는 미해결 문제
- Eve가 양자 컴퓨터를 가지고 있다면? Shor's Algorithm

이산 로그

Computational Complexity

Alice, Bob의 통신 과정을 Eve가 도청했다고 가정

- Eve가 얻은 Q_A, Q_B를 이용하여 z를 계산하기 위해서는 d_A 혹은 d_B를 계산해야 함
- 이는 이산 로그 문제로, 군의 order n에 대해 적어도 $O(\sqrt{n})$의 시간 복잡도를 가짐
  - $O(\sqrt{n})$보다 효율적인 알고리즘이 있는지는 미해결 문제
- Eve가 양자 컴퓨터를 가지고 있다면? Shor's Algorithm

양자 컴퓨터의 상용화 이후를 대비하는 post-quantum cryptography로 타원 곡선 사이의 isogeny에 기반한 암호체계가 제안된 바 있음

새내기의 발표 들어주셔서 감사합니다!

참고 자료 + 참고하면 좋았을... 자료

이인석. *대수학*. 서울: 서울대학교출판부, 2008. Print. 학부 대수학 강의; 2.

- 체, 군 등 대수적 구조 관련 참고

Certicom Research. *SEC 1: Elliptic Curve Cryptography*. Certicom Corp, 2009.

- 타원곡선 암호체계의 표준 문서

Beltrametti, Mauro. et. al. *Lectures on Curves, Surfaces and Projective Varieties : A Classical View of Algebraic Geometry*. Zürich: European Mathematical Society, 2009. Print. EMS Textbooks in Mathematics.

- 도서관에서 빌렸는데 첫페이지 첫단어부터 몰라서 바로 덮었습니다

Luca De Feo. *Mathematics of Isogeny Based Cryptography*. arXiv:1711.04062 [cs.CR]

- Isogeny based Cryptography까지 설명되어 있는 강의노트. 최근 읽어보는 중입니다.