

Isogeny-based cryptography

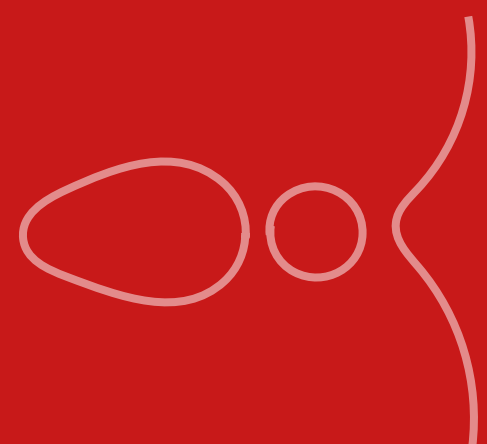
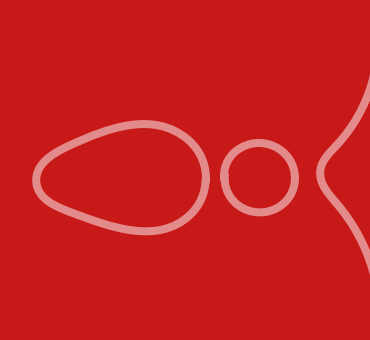
Monika Trimoska

Selected Areas in Cryptology - Part 1

Spring, 2024

TU/e

Elliptic curves



What is an elliptic curve?

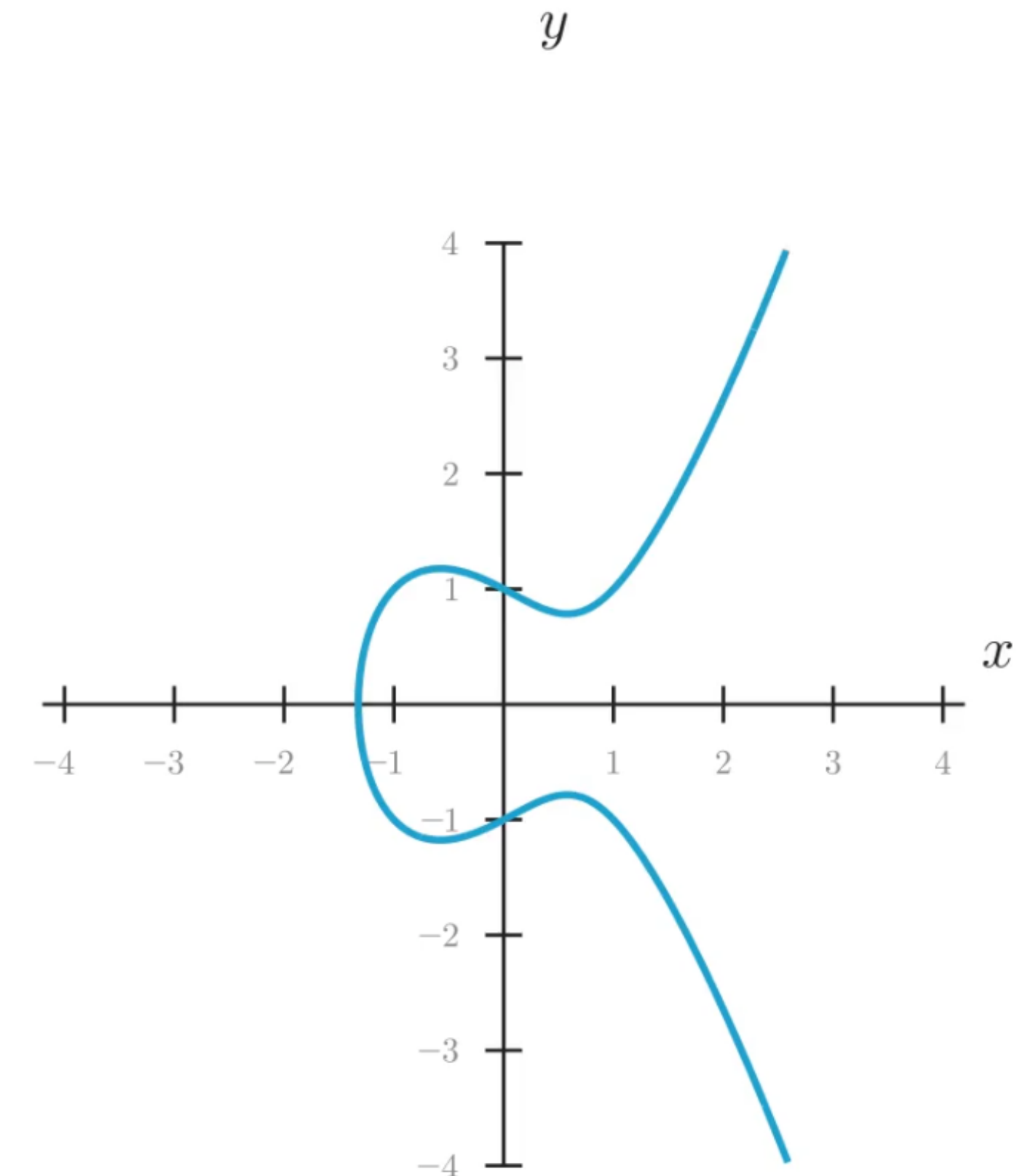
An **elliptic curve** is an algebraic curve that admits an affine equation of the form

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,$$

(general form of a Weierstrass curve)

with $a_i \in k$, where k is the field where the point is defined.

Example. $y^2 = x^3 - x + 1$



→ A **point on E** means that the point (x, y) satisfies the curve equation.

Elliptic curves over \mathbb{F}_q

In cryptography, we use **elliptic curves over finite fields** \mathbb{F}_q , $q = p^k$ (but we draw the figures over \mathbb{R} because it's nicer).

→ We denote by $E(k)$ the set of **k -rational** points on E .

↳ defined over k

Hasse bound

$$\#E(\mathbb{F}_q) \in [q + 1 - 2\sqrt{q}, q + 1 + 2\sqrt{q}]$$

Elliptic curves in cryptography

A curve is

- ▶ **non-singular** (or smooth) if it does not have a singular point.

↳ Jacobi criterion: a point on E is singular if (x, y) also satisfies the two partial derivatives
 $2y + a_1x + a_3 = 0$ and $a_1y = 3x^2 + 2a_2x + a_4$.

- ▶ **supersingular** (also non-singular) if and only if $\#E(\mathbb{F}_p) = p + 1$ (for $p > 3$).

equivalently: iff $E[p] = \{\infty\}$

↳ $E[n] = \{P \in E(\overline{\mathbb{F}}_p) \mid nP = \infty\}$ (the n -torsion group)

Supersingular curves and cyclic groups

- ▶ $E(\mathbb{F}_p) \cong \mathbb{Z}/(p + 1)$

- ▶ $E(\mathbb{F}_{p^2}) \cong \mathbb{Z}/(p + 1) \times \mathbb{Z}/(p + 1)$

More on elliptic curves

- ▶ Short Weierstrass form

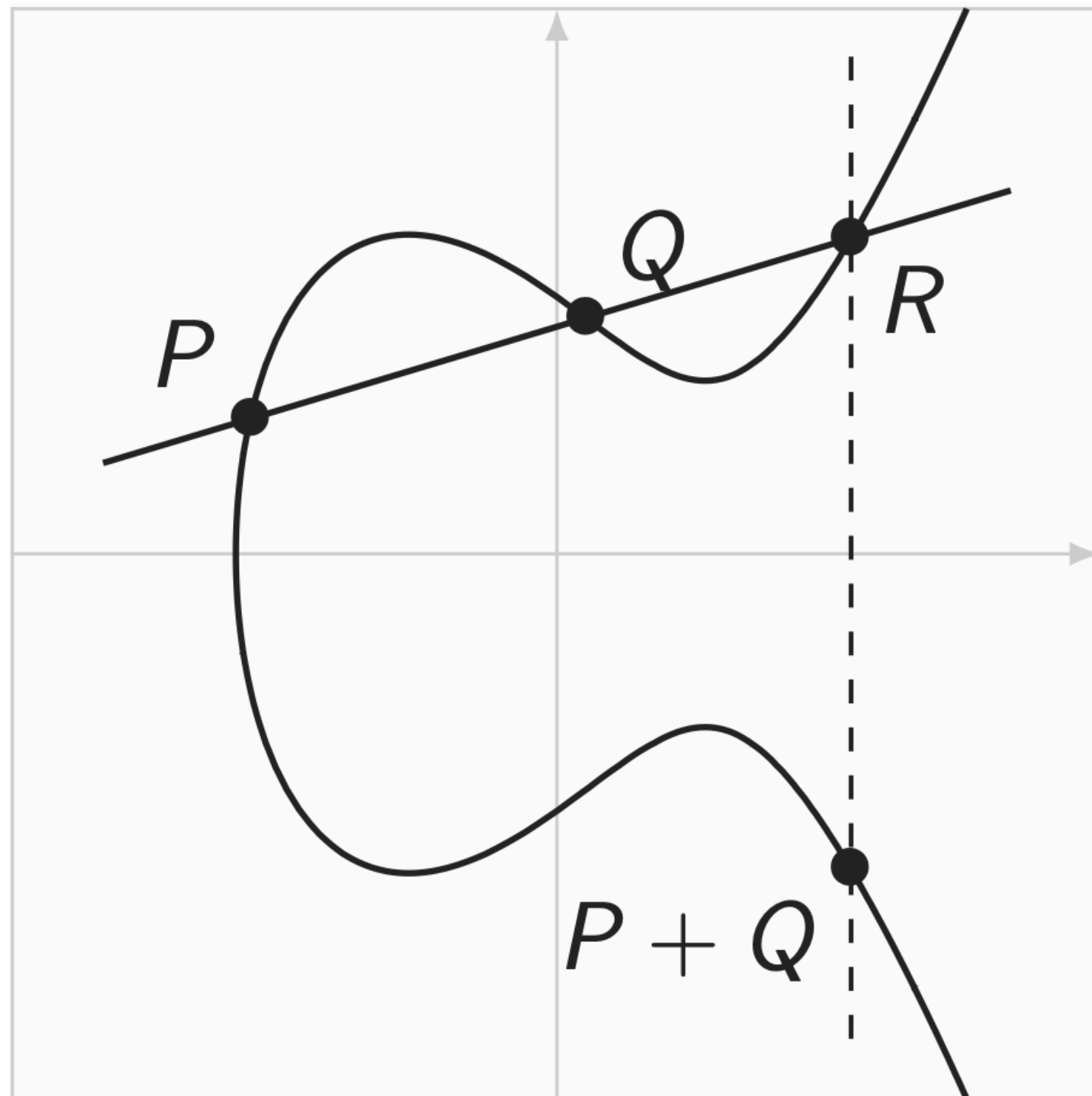
$$y^2 = x^3 + c_4x + c_6$$

→ The curve is non-singular if the **discriminant** $\Delta = 4c_4^3 + 27c_6^2$ is nonzero.

- ▶ The set of points on E with the addition law form a **group**.
- ▶ The **group law** is constructed geometrically.

The geometry of elliptic curves

Adding points on an elliptic curve



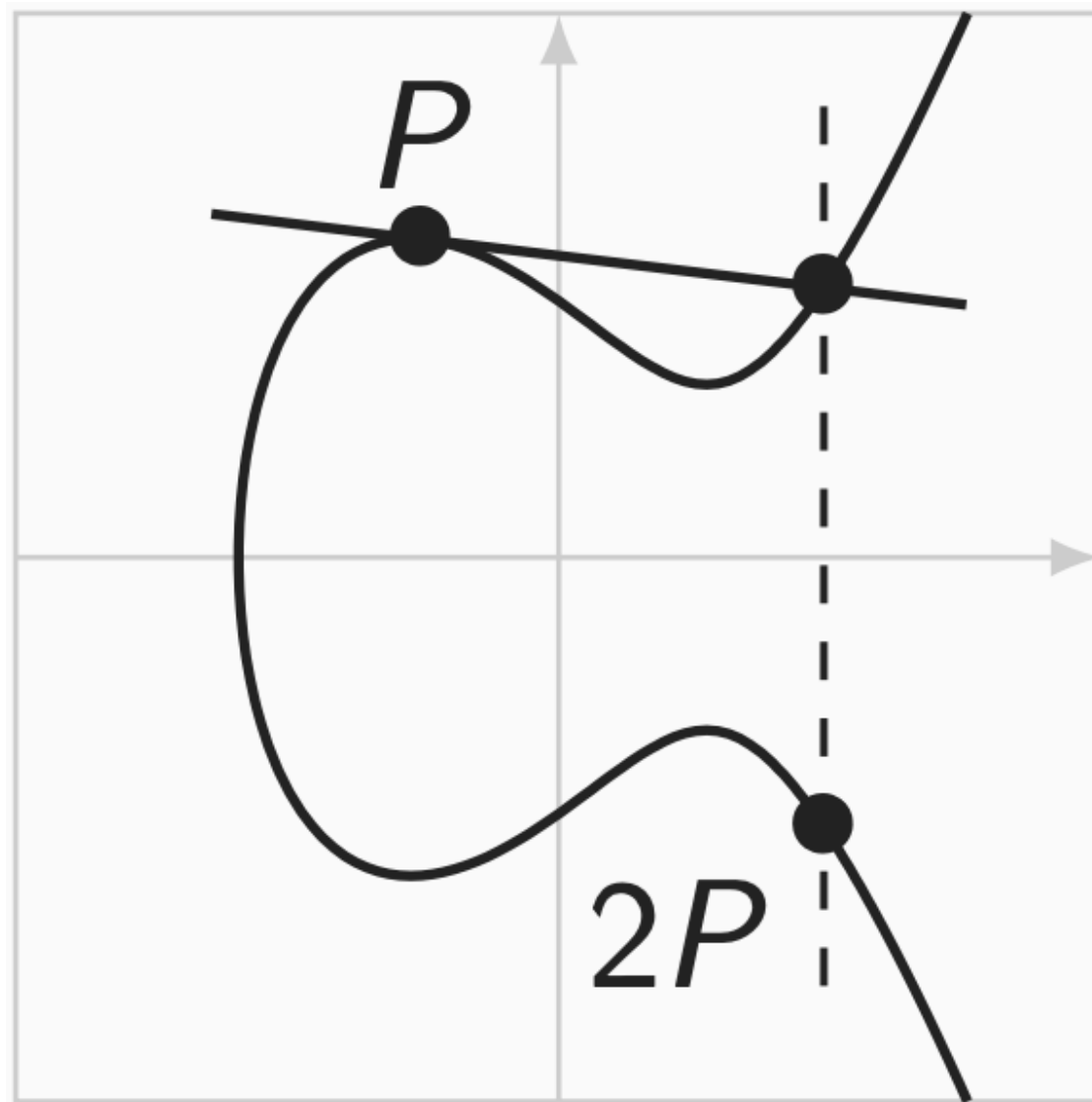
©Eichseder

Addition $P + Q$

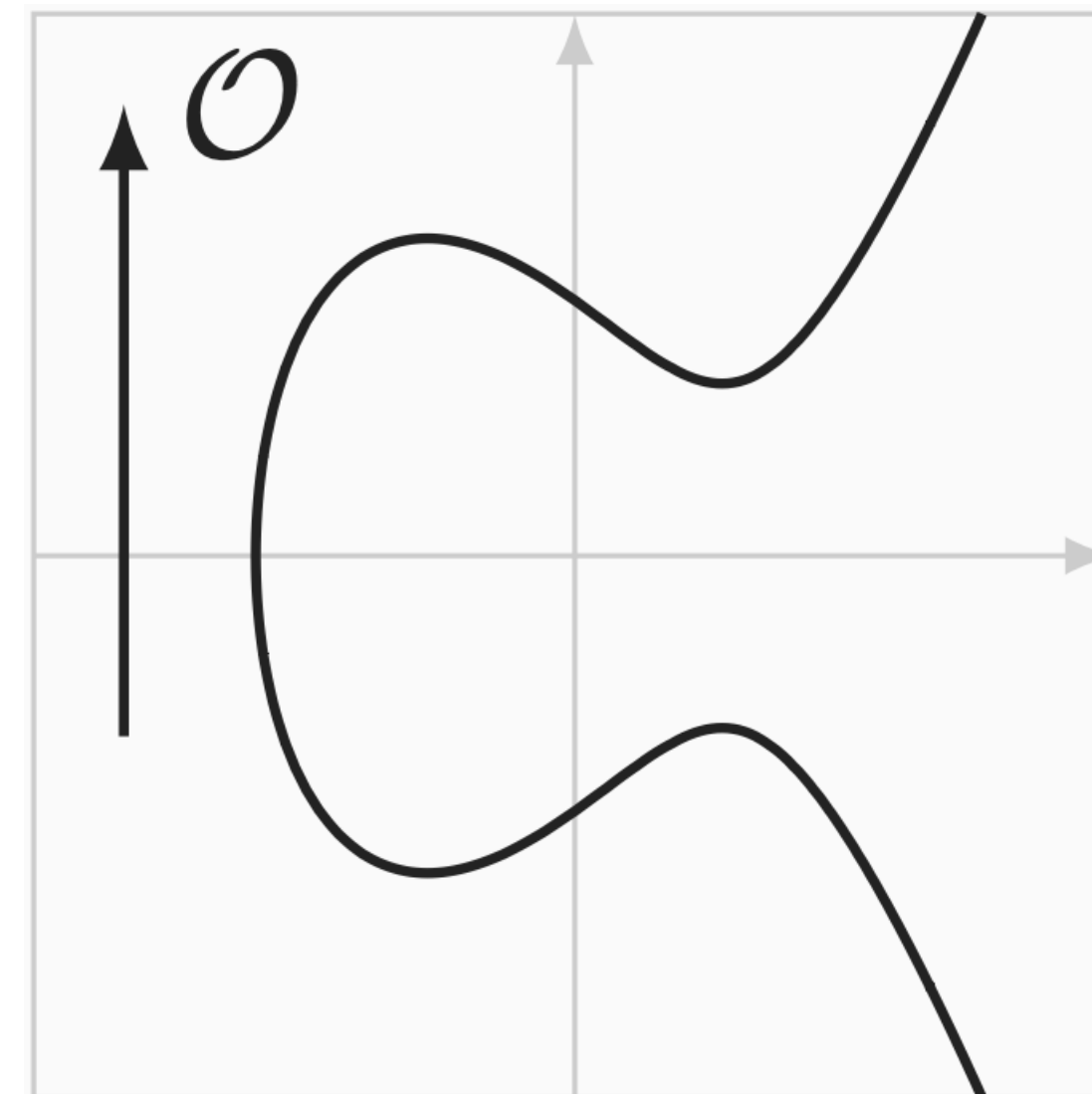
- Draw a line through P and Q .
↪ The line intersects the curve E at a third point R .
- Draw a vertical line through R .
↪ The line intersects E in another point.
- We define that point to be the sum of P and Q .

The geometry of elliptic curves

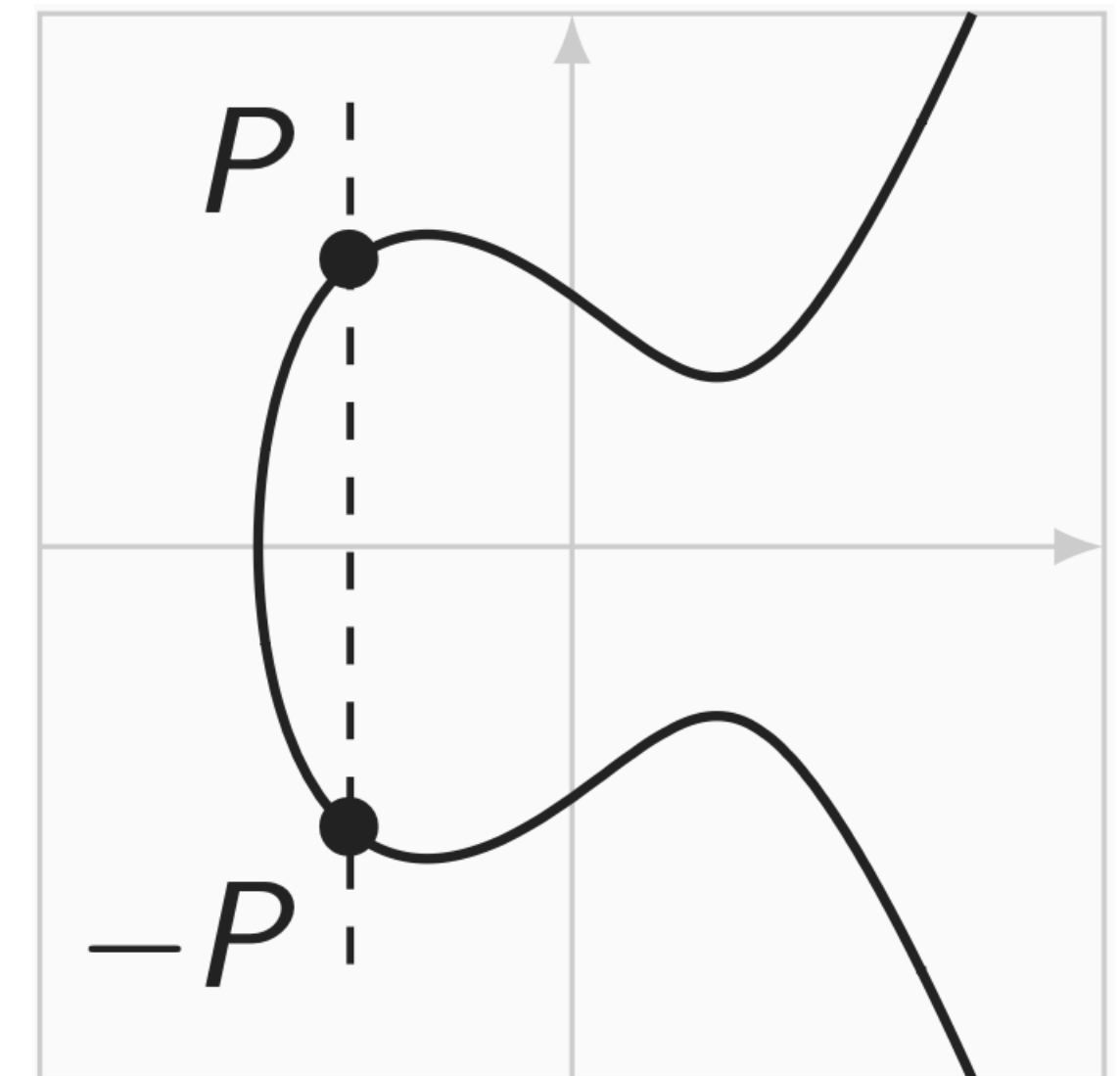
Adding points on an elliptic curve



Doubling $P + P$



Neutral element \mathcal{O}



Inverse element $-P$

- Modify the first step: draw the tangent line to E at P .

The algebra of elliptic curves

The addition law on E has the following properties:

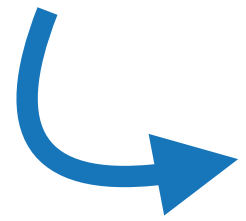
- ▶ $P + \mathcal{O} = P$, for all $P \in E$
- ▶ Let $P \in E$. There is a point on E , denoted by $-P$, satisfying $P + (-P) = \mathcal{O}$.
- ▶ $P + (Q + R) = (P + Q) + R$, for all $P, Q, R \in E$
- ▶ $P + Q = Q + P$, for all $P, Q \in E$

Elliptic curves with points in \mathbb{F}_p are finite abelian groups

- ▶ Closure ✓
- ▶ Associativity ✓
- ▶ Identity element ✓
- ▶ Inverse element ✓
- ▶ Commutativity ✓

The arithmetic of elliptic curves

We can write down explicitly the formulas for the addition law on E .



Let $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$,
then $P_1 + P_2 = (x_3, y_3) = (\lambda^2 - x_1 - x_2, \lambda(x_3 - x_1) + y_1)$, where

$$\lambda = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1}, & \text{when } P_1 \neq P_2 \\ \frac{3x_1^2 + a}{2y_1}, & \text{when } P_1 = P_2. \end{cases}$$

Elliptic curves in SageMath

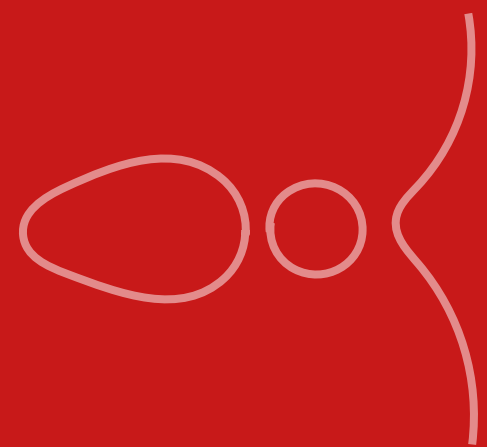
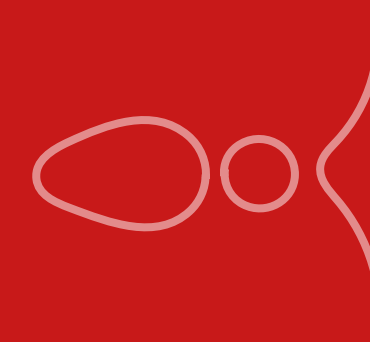
Elliptic curves and the group law

```
p=next_prime(2^8)
K=GF(p)
E=EllipticCurve(K, [0, 0, 0, -1, 1])
print(E)
print("Number of points on E:", E.order())
print("E is supersingular: ", E.order()==p+1)
P=E.random_point()
x=K.random_element()
Q=x*P
print("P: ", P, ", Q: ", Q)
print(P+Q == (x+1)*P)
```

✓ 0.0s

```
Elliptic Curve defined by  $y^2 = x^3 + 256x + 1$  over Finite Field of size 257
Number of points on E: 251
E is supersingular: False
P: (3 : 5 : 1) , Q: (48 : 158 : 1)
True
```

Building crypto from
elliptic curves
(not PQ)



Elliptic curve discrete logarithm problem

The ECDLP problem

Input: Two points $P, Q \in E(\mathbb{F}_q)$.

Question: Find an integer x such that $xP = Q$.

!

We can use the **hardness** of ECDLP only because computing multiples is **easy**.



We can compute mP in $\mathcal{O}(\log m)$ steps by the usual **Double-and-Add Method**.

- ▶ First write $m = m_0 + m_1 \cdot 2 + m_2 \cdot 2^2 + \dots + m_r \cdot 2^r$.
- ▶ Then mP can be computed as $mP = m_0P + m_1 \cdot 2P + m_2 \cdot 2^2P + \dots + m_r \cdot 2^rP$.
- ▶ Requires r doublings (and sums).

Diffie-Hellman key exchange



Alice

Secret a

$$A = aP$$

$$K_a = aB$$

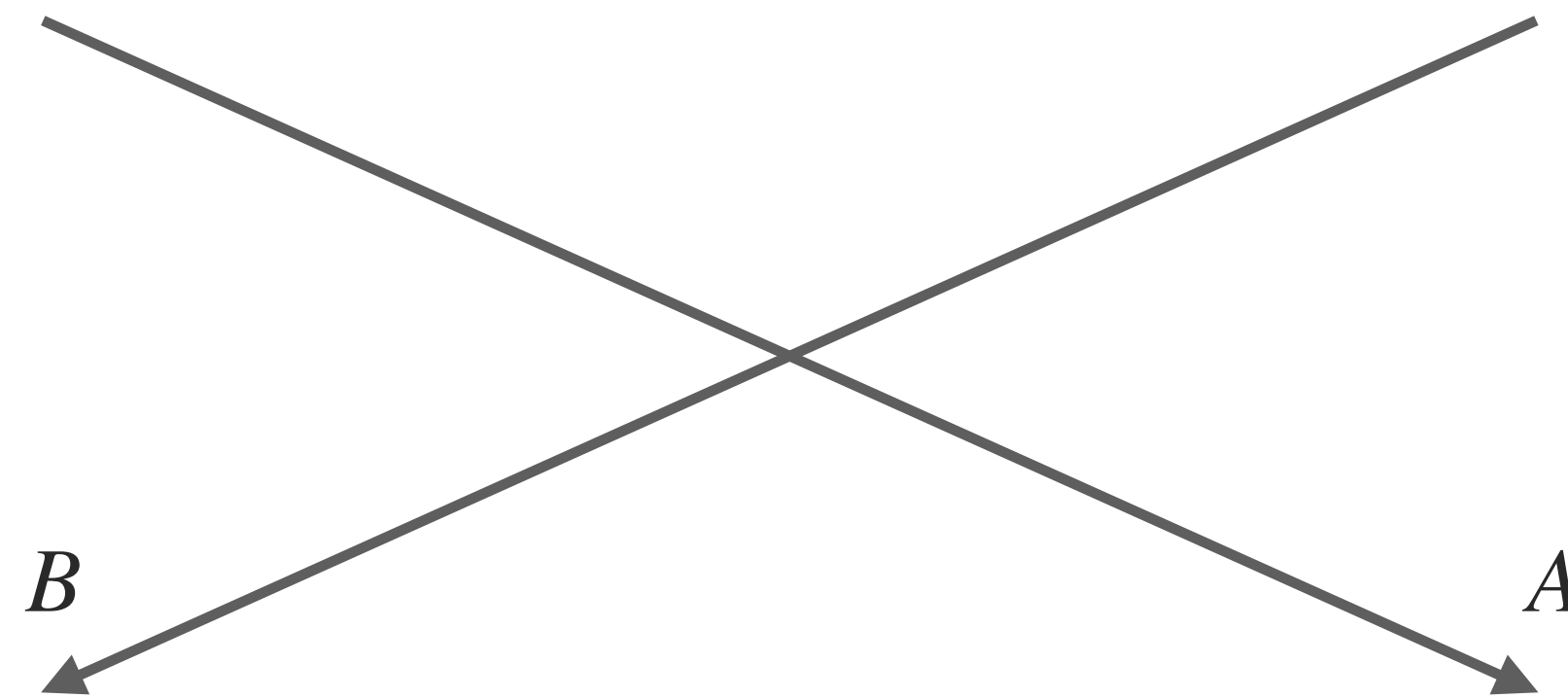


Bob

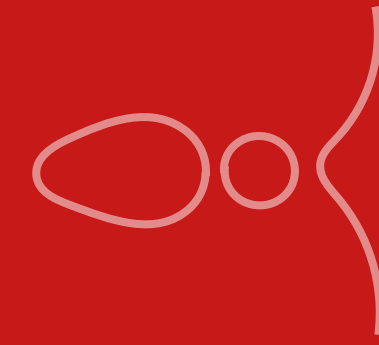
Secret b

$$B = bP$$

$$K_b = bA$$



$$K_a = abP = baP = K_b$$



Isogenies



Isomorphisms

An **isomorphism** is a map between elliptic curves that is defined everywhere, i.e., that is given by polynomials in x and y .

↳ it is a degree-1 isogeny (will make sense on the next slide)
(can be viewed as a change of coordinates)

j -invariant

- ▶ An invariant under isomorphisms.
- ▶ For a curve in short Weierstrass form $y^2 = x^3 + c_4x + c_6$, we have

$$j = 1728 \cdot 4c_4^3 / (4c_4^3 + 27c_6^2)$$

Isogenies

An **isogeny** φ of elliptic curves is a non-zero map $E \rightarrow E'$ that is

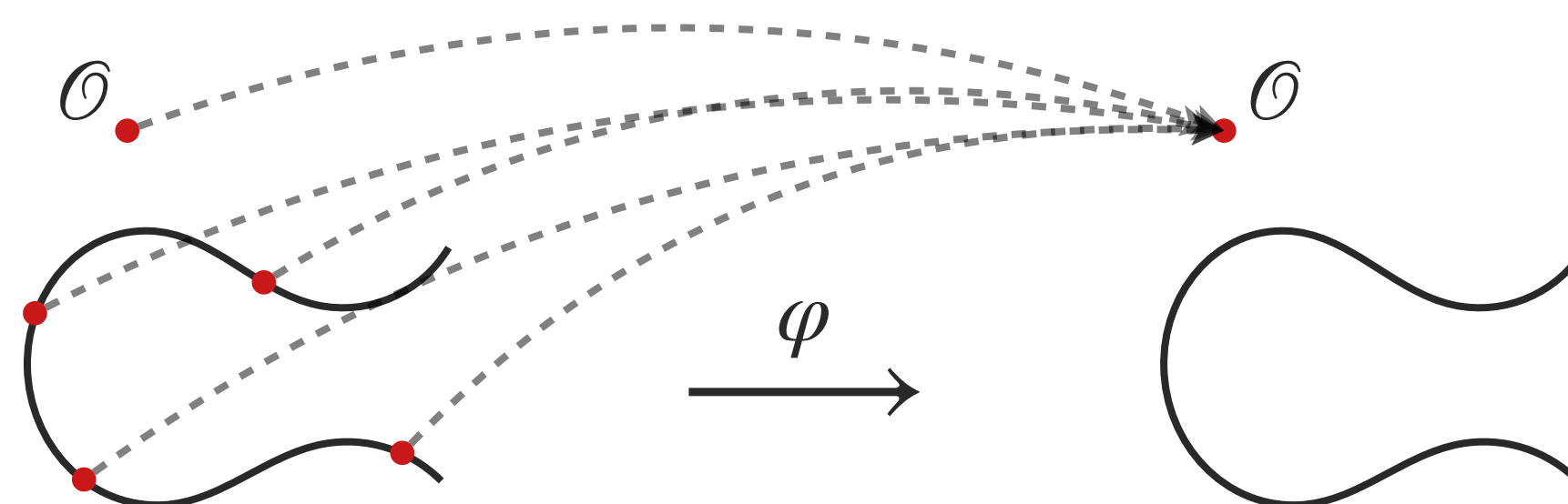
- ▶ given by **rational functions**
- ▶ that is a **group homomorphism**

$$\varphi(P + Q) = \varphi(P) + \varphi(Q)$$

$$\frac{f(x, y)}{g(x, y)}, \text{ where } f, g \text{ are polynomials}$$

→ An isogeny is uniquely defined by its **kernel**: $\{P \in E \mid \varphi(P) = \mathcal{O}_{E'}\}$.

→ The **degree** of a (separable) isogeny is the size of its kernel.



Isogenies

Example.

$$(x, y) \mapsto \left(\frac{x^3 - 4x^2 + 30x - 12}{(x - 2)^2}, \frac{x^3 - 6x^2 - 14x + 35}{(x - 2)^3} \cdot y \right)$$

defines a degree-3 isogeny of the elliptic curves

$$\{y^2 = x^3 + x\} \rightarrow \{y^2 = x^3 - 3x + 3\}$$

over \mathbb{F}_{71} . Its kernel is $\{(2, 9), (2, -9), \mathcal{O}\}$.

ℓ -isogeny:

- ▶ $x \rightarrow \frac{f(x)}{g(x)}$, with $\deg(f) = \ell$, $\deg(g) = \ell - 1$
- ▶ $y \rightarrow \dots$

Computing isogenies

*We consider only supersingular curves from now on.

→ **Goal:** Compute an ℓ -isogeny from E .

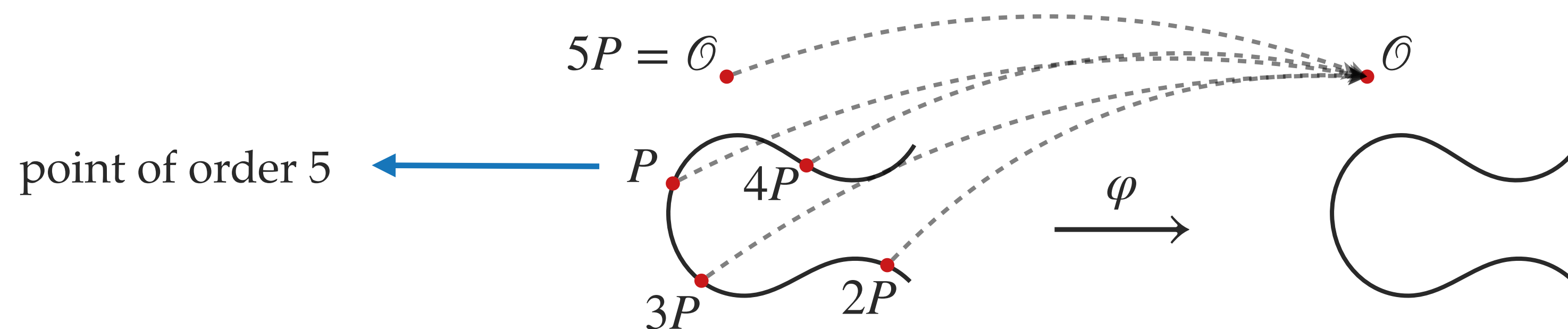
Find a point P on E of order ℓ

↳ Take the cyclic group generated by P

↳ Obtain a subgroup of order ℓ

↳ Obtain a kernel of an ℓ -isogeny

↳ Compute an ℓ -isogeny



Vélu's formulas

→ For any **finite** subgroup G of E , there exists a **unique** (up to isomorphism) separable isogeny $\varphi_G : E \rightarrow E'$ with kernel G .

Vélu '71

- ▶ Formulas for computing E' and evaluating φ_G at a point.
- ▶ Complexity: $\Theta(\#G)$ → only suitable for small degrees.

Let P have prime order ℓ on E_A .

For $1 \leq i < \ell$ let x_i be the x -coordinate of iP .

Let

$$\tau = \prod_{i=1}^{\ell-1} x_i, \quad \sigma = \sum_{i=1}^{\ell-1} \left(x_i - \frac{1}{x_i} \right), \quad f(x) = x \prod_{i=1}^{\ell-1} \frac{xx_i - 1}{x - x_i}.$$

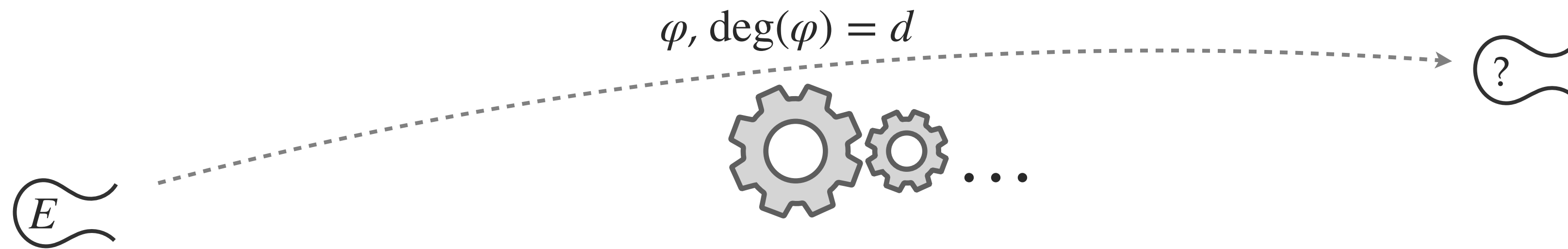
Then the ℓ -isogeny with kernel $\langle P \rangle$ is given by

$$\varphi_\ell : E_A \rightarrow E_B, (x, y) \mapsto (f(x), c_0 y f'(x))$$

where $B = \tau(A - 3\sigma)$, and $c_0^2 = \tau$.

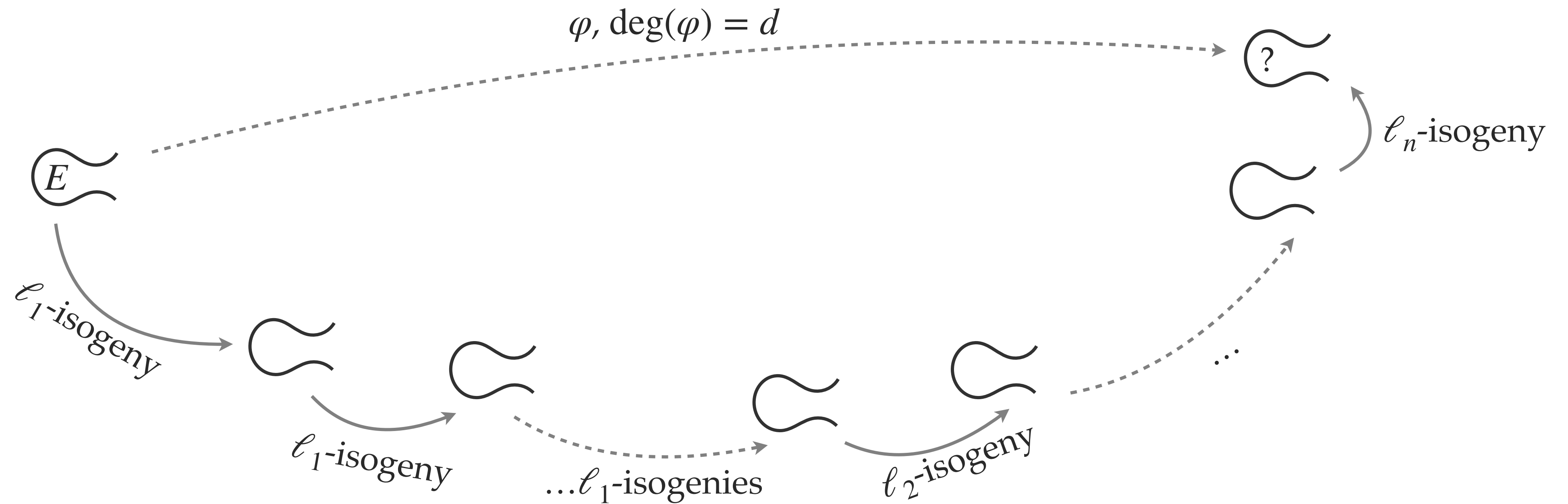
Composing isogenies

→ Goal: Compute a d -isogeny from E .



Composing isogenies

→ Goal: Compute a d -isogeny from E , with d a smooth integer ($d = \ell_1^{e_1} \cdot \ell_2^{e_2} \cdots \ell_n^{e_n}$).



Isogenies in SageMath

Computing isogenies

```
p=139
A=0
E=EllipticCurve(GF(p), [0, A, 0, 1, 0])
assert E.order()==p+1 #check that it is a supersingular curve
print("We can compute isogenies of the following degrees:", factor((p+1)/4))
P=E.random_point()
while P.order().is_prime() == False:
    P=E.random_point()
print("We will compute an isogeny of degree", P.order())
print(E.montgomery_model()) #needs Sage 10.3
phi=E.isogeny(P)
print(phi)
E2=phi.codomain()
print(E2.montgomery_model()) #needs Sage 10.3
```

✓ 0.0s

We can compute isogenies of the following degrees: 5 * 7

We will compute an isogeny of degree 5

Elliptic Curve defined by $y^2 = x^3 + x$ over Finite Field of size 139

Isogeny of degree 5 from Elliptic Curve defined by $y^2 = x^3 + x$ over Finite Field of size 139 to Elliptic Curve defined by $y^2 = x^3 + 72x + 30$ over Finite Field of size 139

Elliptic Curve defined by $y^2 = x^3 + 126x^2 + x$ over Finite Field of size 139

Endomorphism rings

Dual isogeny

- ▶ For isogeny $\varphi : E \rightarrow E'$ there exists a unique **dual isogeny** $\hat{\varphi} : E' \rightarrow E$.
- ▶ The composition $\hat{\varphi} \circ \varphi$ is **the multiplication-by- d map** on E and $\varphi \circ \hat{\varphi}$ the multiplication-by- d map on E' , where $d = \deg(\varphi) = \deg(\hat{\varphi})$.

The multiplication-by- d map

- ▶ The multiplication-by- d map $[d] : E \rightarrow E$ is a **degree- d^2** isogeny from E to E .
 - ▶ Its kernel is $E[d] \cong \mathbb{Z}/d \times \mathbb{Z}/d$.
- It is an endomorphism.

End(E)

- ▶ An **endomorphism** is an isogeny from a curve E to itself.
- ▶ The set of endomorphisms forms a ring $\text{End}(E)$ under $+$ and \circ .

Hard problems and reductions

The isogeny path problem

Input: Two supersingular curves E and E' .
Question: Find an isogeny φ from E to E' .

The **decisional** problem is easy
(when point counting is easy)

Tate's theorem: E and E' are isogenous over \mathbb{F}_p if and only if $\#E(\mathbb{F}_p) = \#E'(\mathbb{F}_p)$.

The EndRing problem

Input: A super singular curve E .
Question: Find a basis of $\text{End}(E)$.

Isogeny graphs

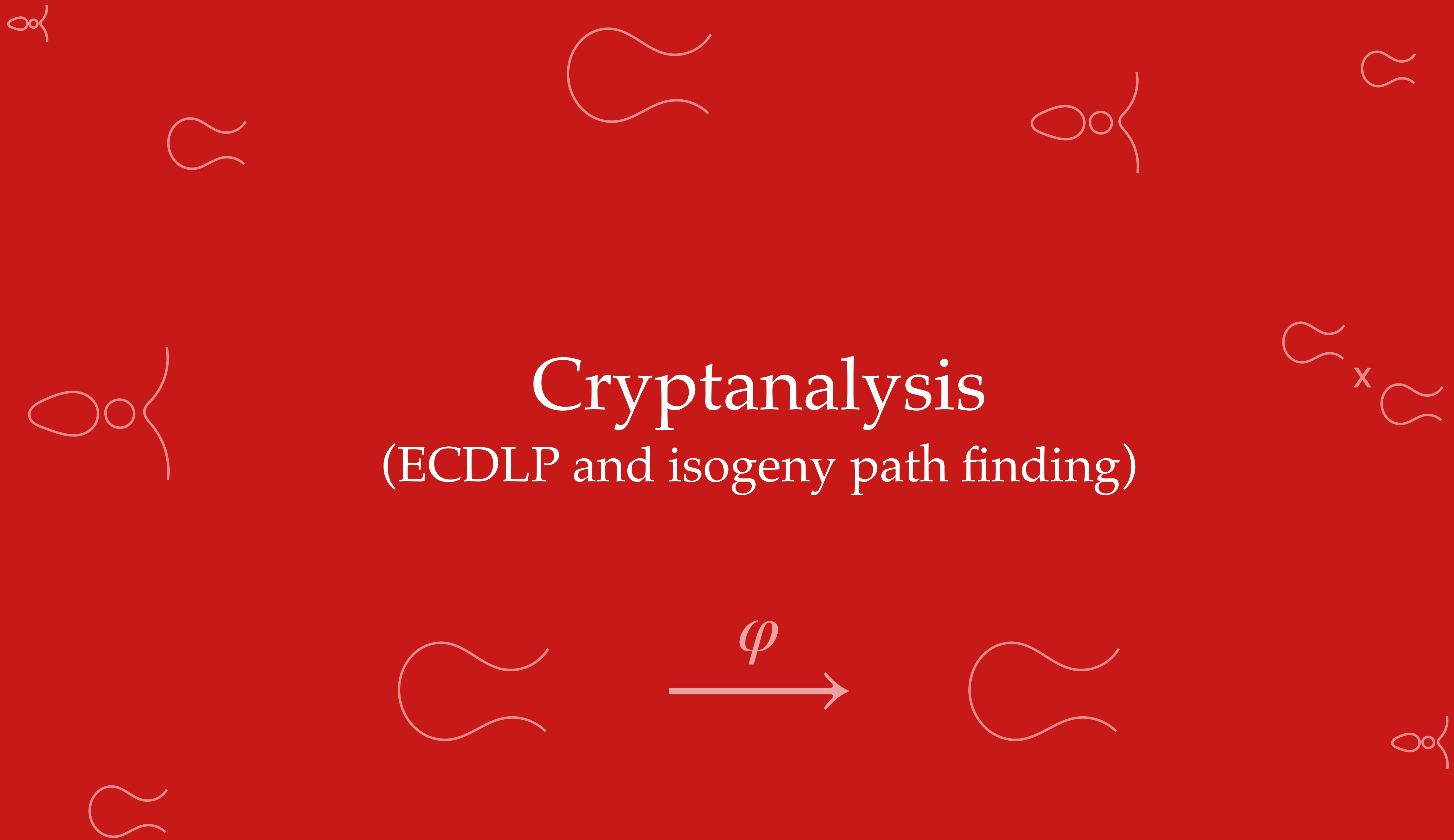
- ▶ **Vertices** are \mathbb{F}_p -isomorphism classes of supersingular elliptic curves.
- ▶ **Edges** are prime-degree isogenies between them.

Cryptanalysis

(ECDLP and isogeny path finding)




$$\mathcal{C}_1 \times \mathcal{C}_2$$



Cryptanalysis

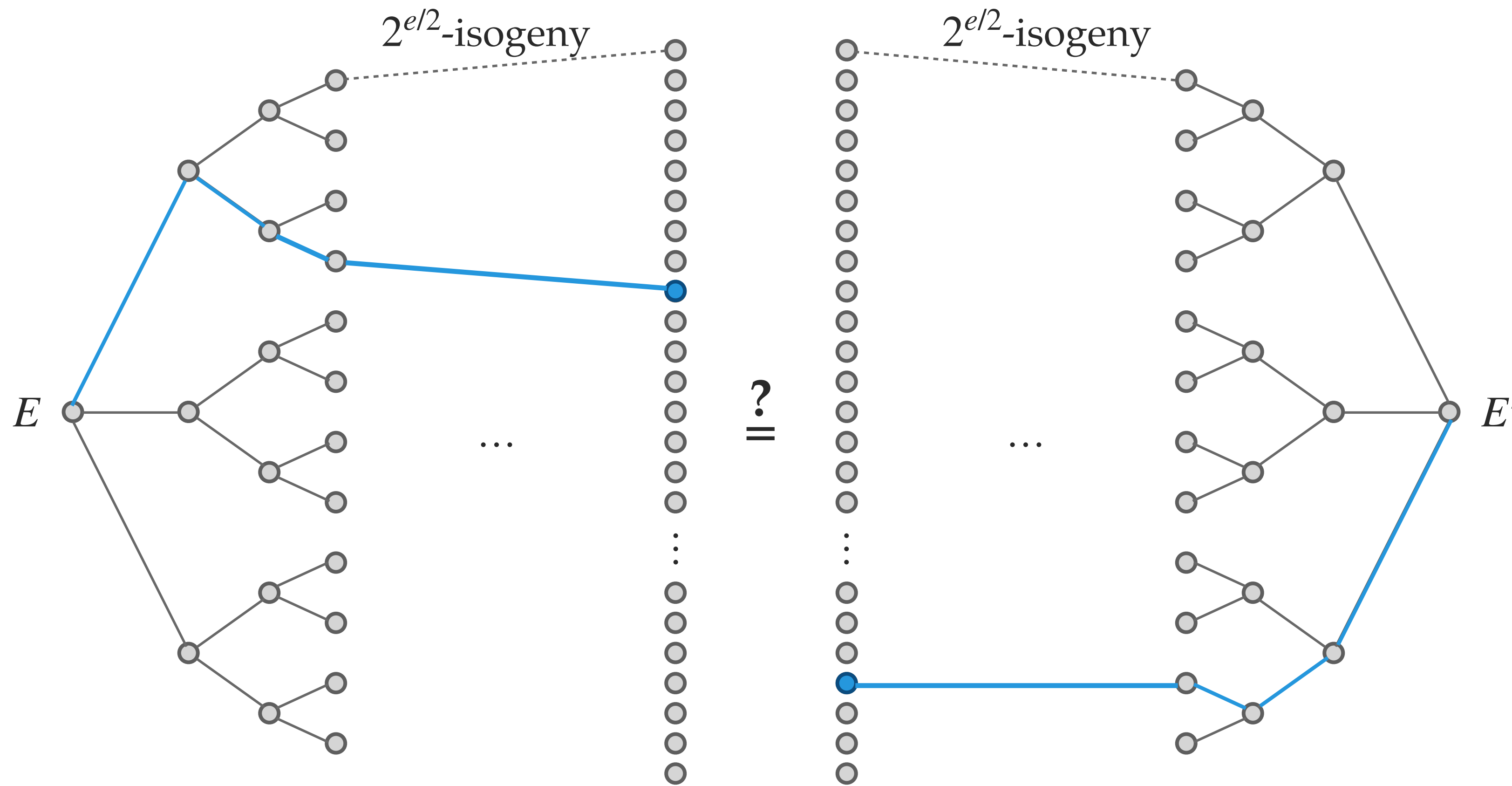
Generic attacks are all in $\tilde{O}(N^{\frac{1}{2}})$, where N is the size of the search space.

- 
- ▶ $\#E(\mathbb{F}_q)$ (for ECDLP)
 - ▶ Nb. of isogenies from E of the fixed degree (for fixed-degree isogeny path finding)
 - ▶ etc.

- ▶ Meet-in-the-middle
- ▶ Parallel Collision Search (vOW)
- ▶ Delfs–Galbraith

Meet-in-the-middle

Example. Goal: Find a 2^e -isogeny from E to E' .



 More details in the assignment.

Collision search

What is a **collision**? Why does a collision help us solve the (EC)DLP?

 Having two different linear combinations of a random point $R \in E(\mathbb{F}_q)$

$$R = aP + bQ$$

$$R = a'P + b'Q$$

we infer that

$$aP + bQ = a'P + b'Q$$

$$(a - a')P = (b' - b)Q$$

and we compute

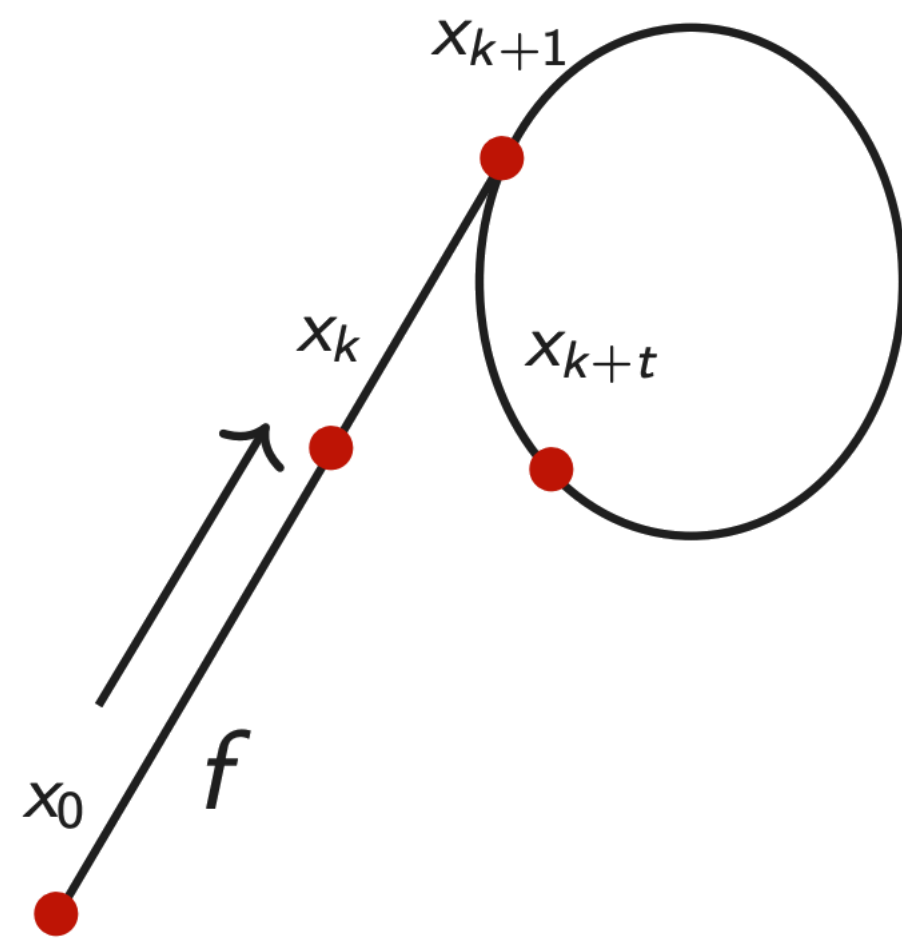
$$x = \frac{a - a'}{b' - b} \pmod{N}.$$

Collision search

Collision

Given a random map $f: S \rightarrow S$ on a finite set S of cardinality N , we call collision any pair R, R' of elements in S such that $f(R) = f(R')$.

Pollard's Rho method



- ▶ Ideally, f is a random mapping.
- ▶ Expected number of steps until the collision is found

$$\sqrt{\frac{\pi N}{2}}$$

Collision search

$$f(R) = \begin{cases} R + P & \text{if } R \in S_1 \\ 2R & \text{if } R \in S_2 \\ R + Q & \text{if } R \in S_3, \end{cases}$$

Property of f

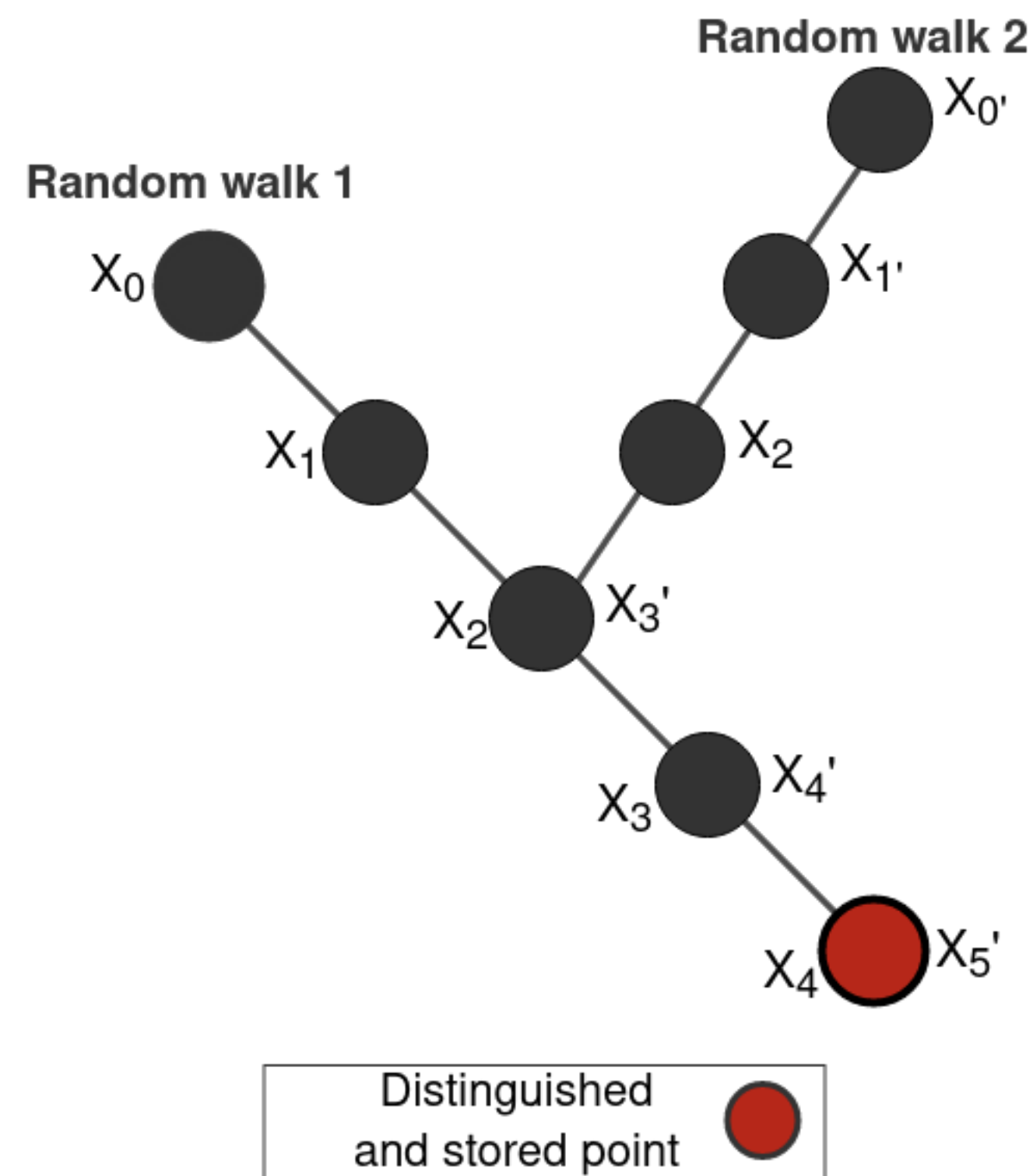
Input $(aP + bQ) \rightarrow$ Output $(a'P + b'Q)$.

(If the input of f is linear combination of P and Q , the output of f is also a linear combination of P and Q .)

Intuitively:

- ▶ Start from $R = aP + bQ$ for some random a and b .
- ▶ *Walk* the random walk until we find the same point twice.
 - ↪ To discover the collision, we need to store *all** the points that we compute.

Parallel Collision Search

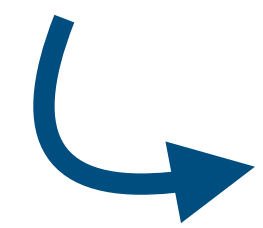


- ▶ Proposed by van Oorschot & Wiener (1996).
- ▶ **Distinguished points**: a set of points having an easily testable property.
ex. The x -coordinate has 3 trailing zero bits:
10101101**000**.
- ▶ Only distinguished points are stored in memory.
- ▶ θ - the **proportion** of distinguished points in a set S .
- ▶ Complexity? How many points do we **expect** to compute (store) before a collision is found?

↪ The Birthday paradox

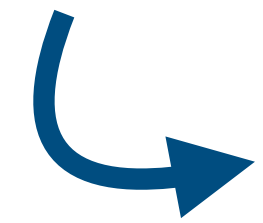
↪ (recall) $\sim \sqrt{N}$

PCS for isogenies



Yes, but it becomes a multi-collision search (finding the **golden** collision).

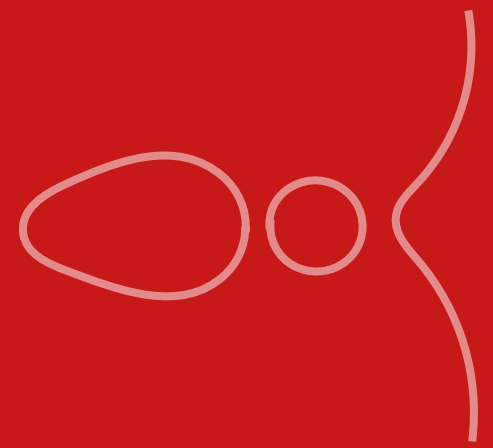
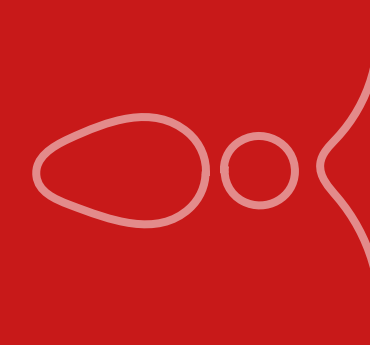
Even less memory



Delfs–Galbraith algorithm.

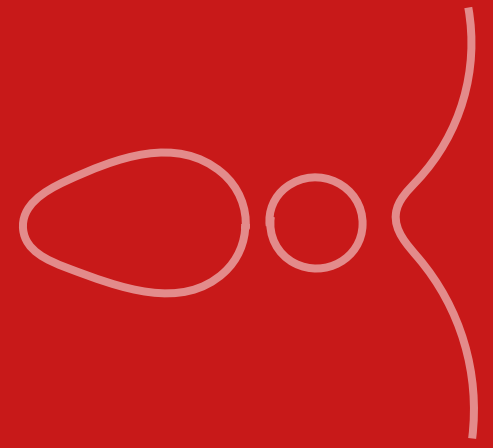
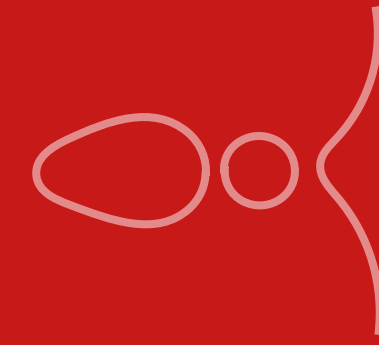
- ▶ Only for the isogeny setting.
- ▶ Negligible space requirements.

Building crypto from
elliptic curves
(not PQ)



Building crypto from
~~elliptic curves~~ isogenies
(PQ)



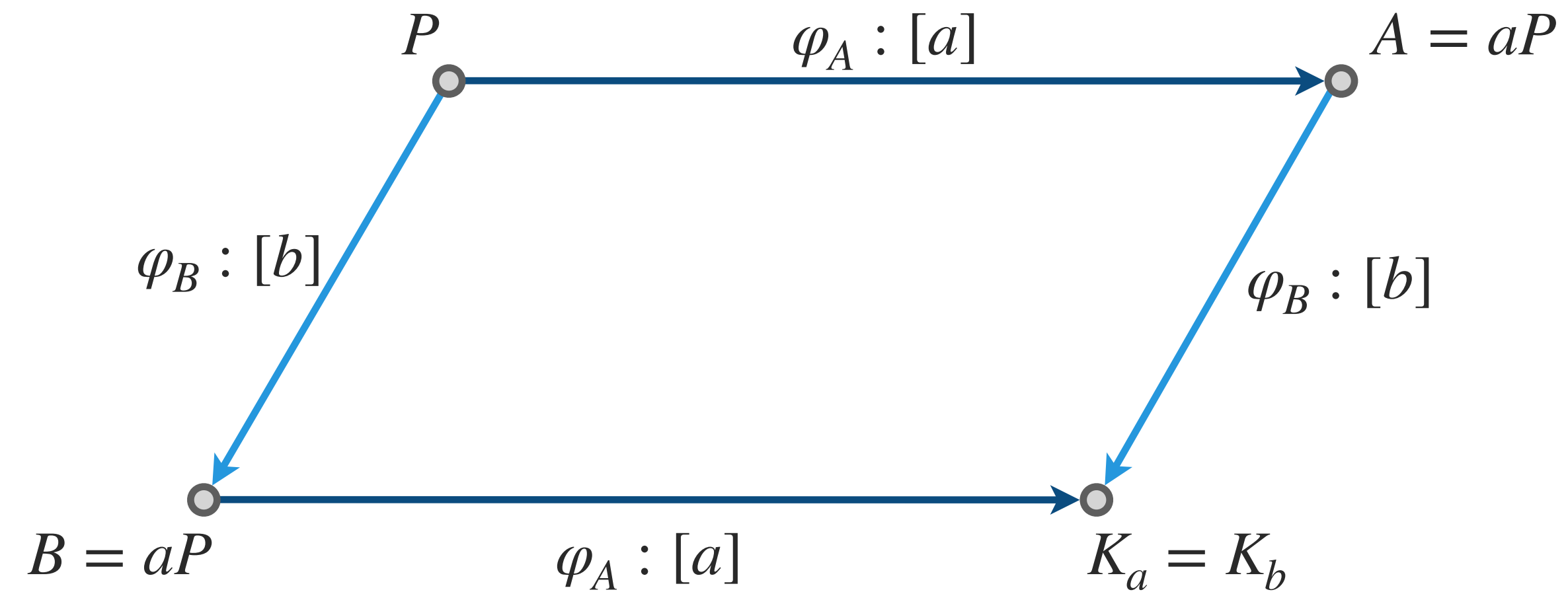


CSIDH

DH key exchange on graphs

Imagine *the dlog graph*

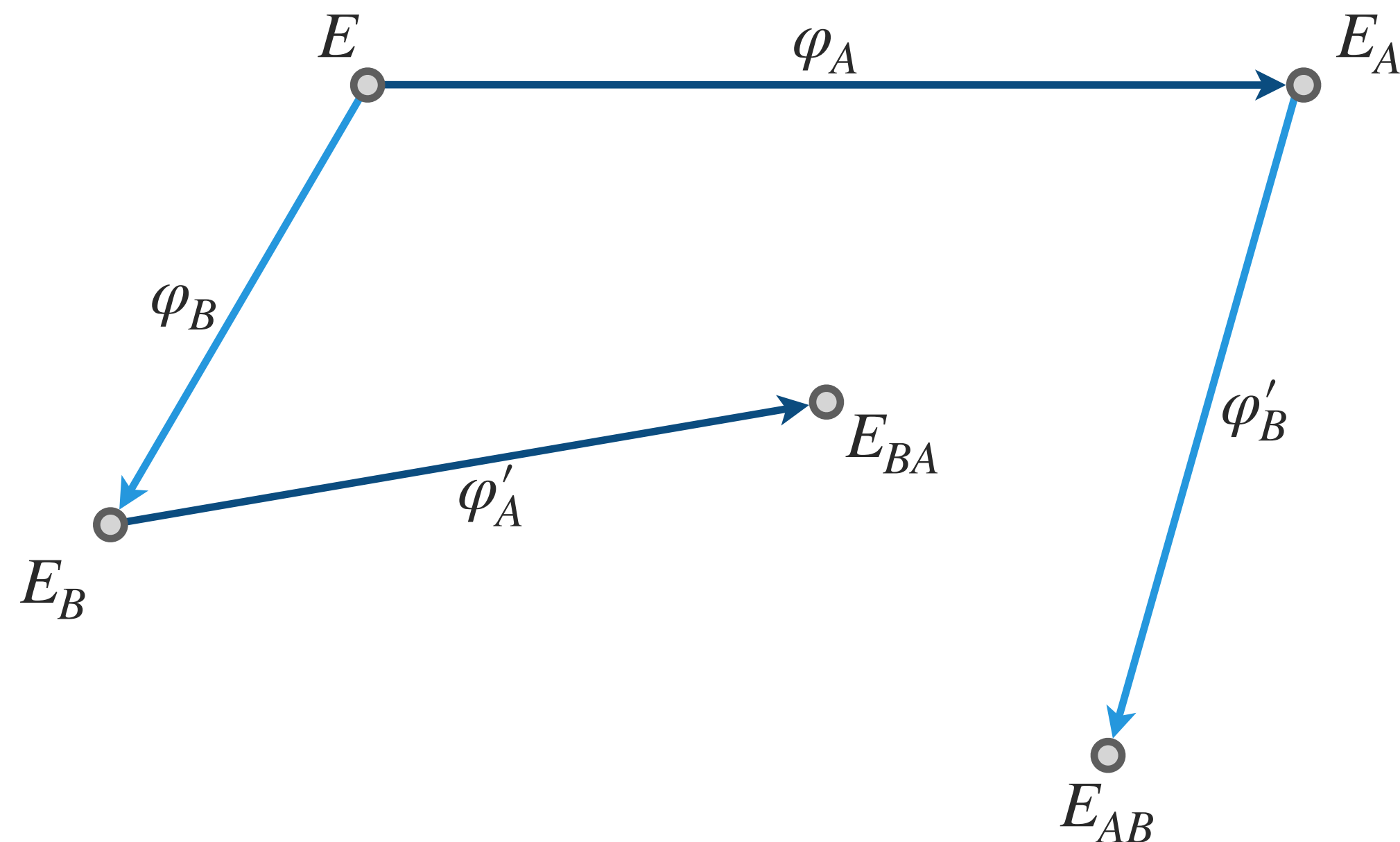
- ▶ **Vertices** are points on E .
- ▶ **Edges** are multiplication-by- i maps.



DH key exchange on isogeny graphs?

Isogeny graphs

- ▶ **Vertices** are isomorphism classes of supersingular elliptic curves.
- ▶ **Edges** are prime-degree isogenies between them.



→ Walking on the isogeny graph is not **commutative** (a priori).

↪ Alice & Bob do not end up on the same vertex (isomorphism class).

Commutative group action

- **Fundamental theorem of cyclic groups** ^{*additive notation.}

Every subgroup of a cyclic group $G = \langle P \rangle$ is cyclic.
Moreover, if $\#G = N$, then the order of any subgroup of G is a divisor of N , and,
for each positive divisor k of N , the group G has exactly one subgroup of order k : namely, $\langle [N/k]P \rangle$.

- **Supersingular curves and cyclic groups (recall)**

- ▶ $E(\mathbb{F}_p) \cong \mathbb{Z}/(p+1)$
- ▶ $E(\mathbb{F}_{p^2}) \cong \mathbb{Z}/(p+1) \times \mathbb{Z}/(p+1)$

$$\text{Let } p = 4 \cdot \ell_1 \cdots \ell_n - 1.$$

There is exactly one \mathbb{F}_p -rational ℓ -isogeny from each E .

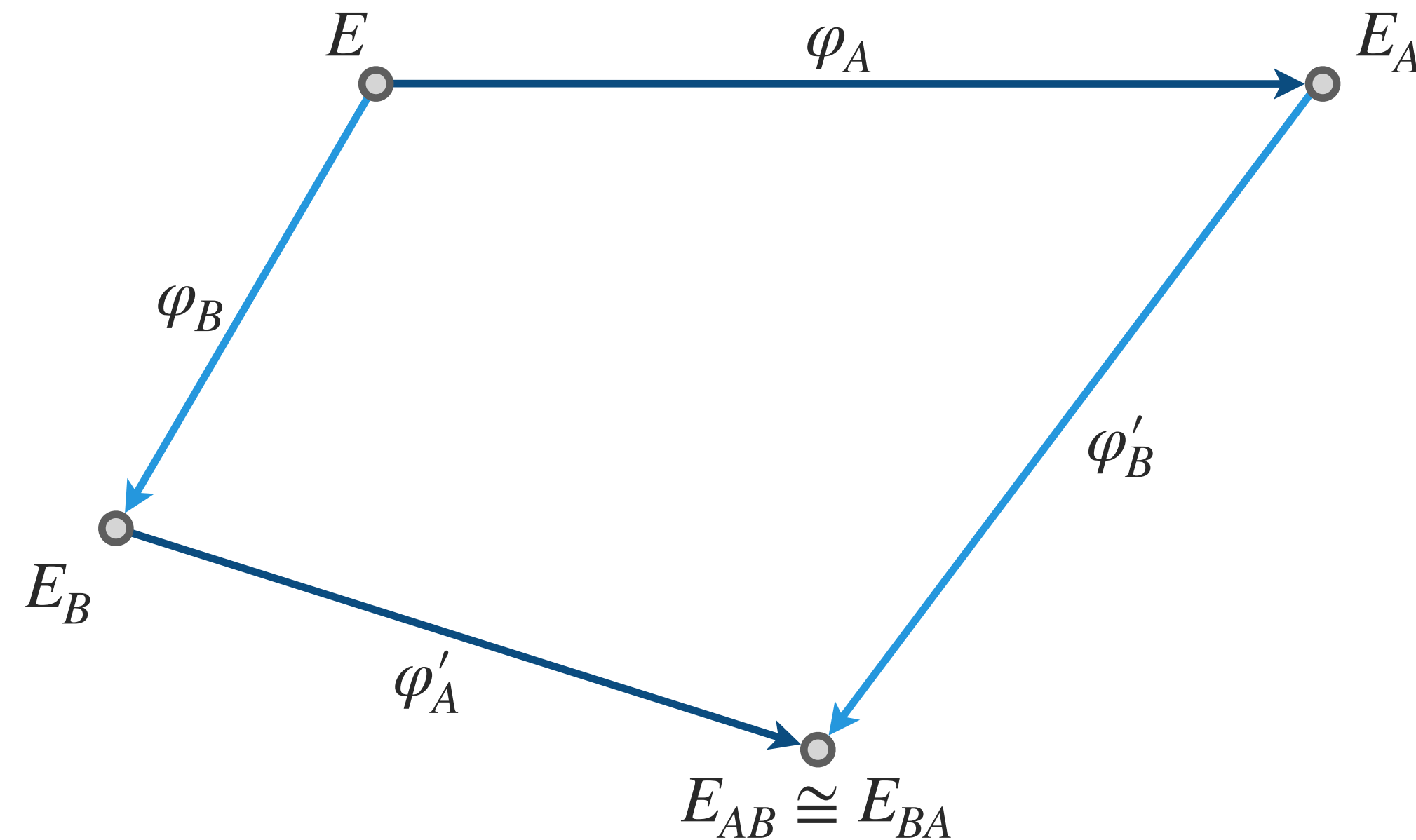
There are exactly $(\ell + 1)$ \mathbb{F}_{p^2} -rational ℓ -isogenies from each E .

▶ Taking the $E(\mathbb{F}_p)$ isogeny graph will give us a commutative group action.

DH key exchange on isogeny graphs

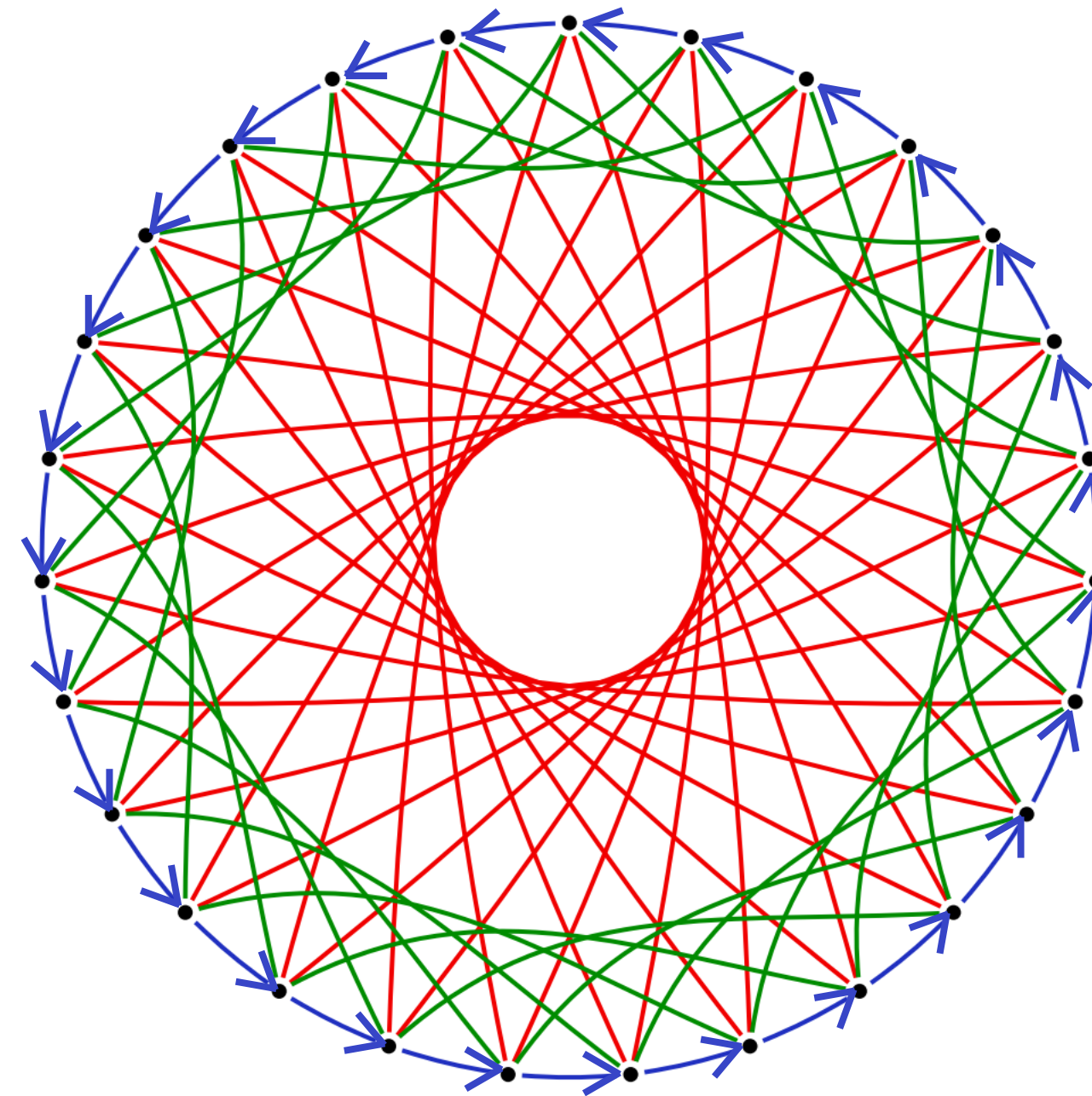
Isogeny graphs $E(\mathbb{F}_p)$ with $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ a prime.

- ▶ Vertices are \mathbb{F}_p -isomorphism classes of supersingular elliptic curves.
- ▶ Edges are prime-degree isogenies between them.



The CSIDH graph

Example. Let $p = 4 \cdot 3 \cdot 5 \cdot 7 - 1$.



- 3-isogeny
- 5-isogeny
- 7-isogeny

Quadratic twists

E'/k is a **twist** of elliptic curve E/k if E' is isomorphic to E over \bar{k} .

For $E : y^2 = x^3 + Ax^2 + x$ over \mathbb{F}_p with $p \equiv 3 \pmod{4}$
 $E' : -y^2 = x^3 + Ax^2 + x$ is isomorphic to E via

$$(x, y) \mapsto (x, \sqrt{-1}y).$$

This map is defined over \mathbb{F}_{p^2} , so this is a **quadratic twist**.

E' is not in Weierstrass form (does not have the right shape).

E' is isomorphic to $E'' : y^2 = x^3 - Ax^2 + x$ via $(x, y) \mapsto (-x, y)$ over \mathbb{F}_p .

Each $x \in \mathbb{F}_p$ satisfies one of

- ▶ $x^3 + Ax^2 + x$ is a square in \mathbb{F}_p , thus there are two points $(x, \pm\sqrt{x^3 + Ax^2 + x})$ in $E(\mathbb{F}_p)$.
- ▶ $x^3 + Ax^2 + x$ is not a square in \mathbb{F}_p , thus there are two points $(x, \pm\sqrt{-(x^3 + Ax^2 + x)})$ in $E'(\mathbb{F}_p)$.
- ▶ $x^3 + Ax^2 + x = 0$, thus $(x, 0)$ is a point in $E(\mathbb{F}_p)$ and in $E'(\mathbb{F}_p)$.

©Lange

Quadratic twists in SageMath

Quadratic twists

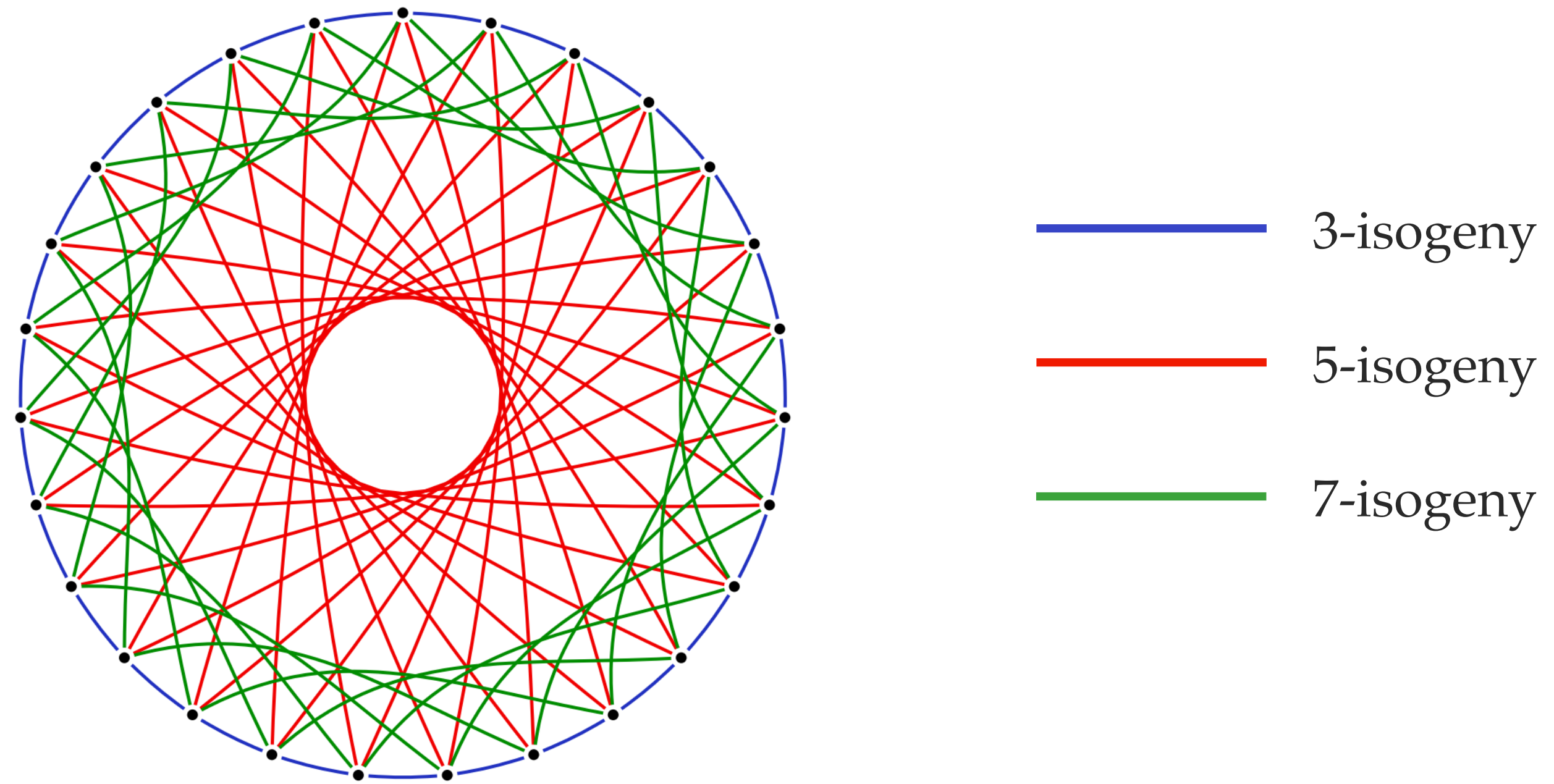
```
p=419
Fp=GF(p)
Fp2=GF(p^2)
E=EllipticCurve(Fp, [0, 410, 0, 1, 0])
assert E.order()==p+1 #check that it is a supersingular curve
E_t=E.quadratic_twist()
print("The quadratic twist of E_", E.montgomery_model().a2(), "is E_", E_t.
montgomery_model().a2())
print("Indeed, -", E.montgomery_model().a2(), "is ", -Fp(E.montgomery_model().a2
()), "over ", Fp)
```

✓ 0.0s

```
The quadratic twist of E_ 410 is E_ 9
Indeed, - 410 is 9 over Finite Field of size 419
```

The CSIDH graph

Example. Let $p = 4 \cdot 3 \cdot 5 \cdot 7 - 1$.



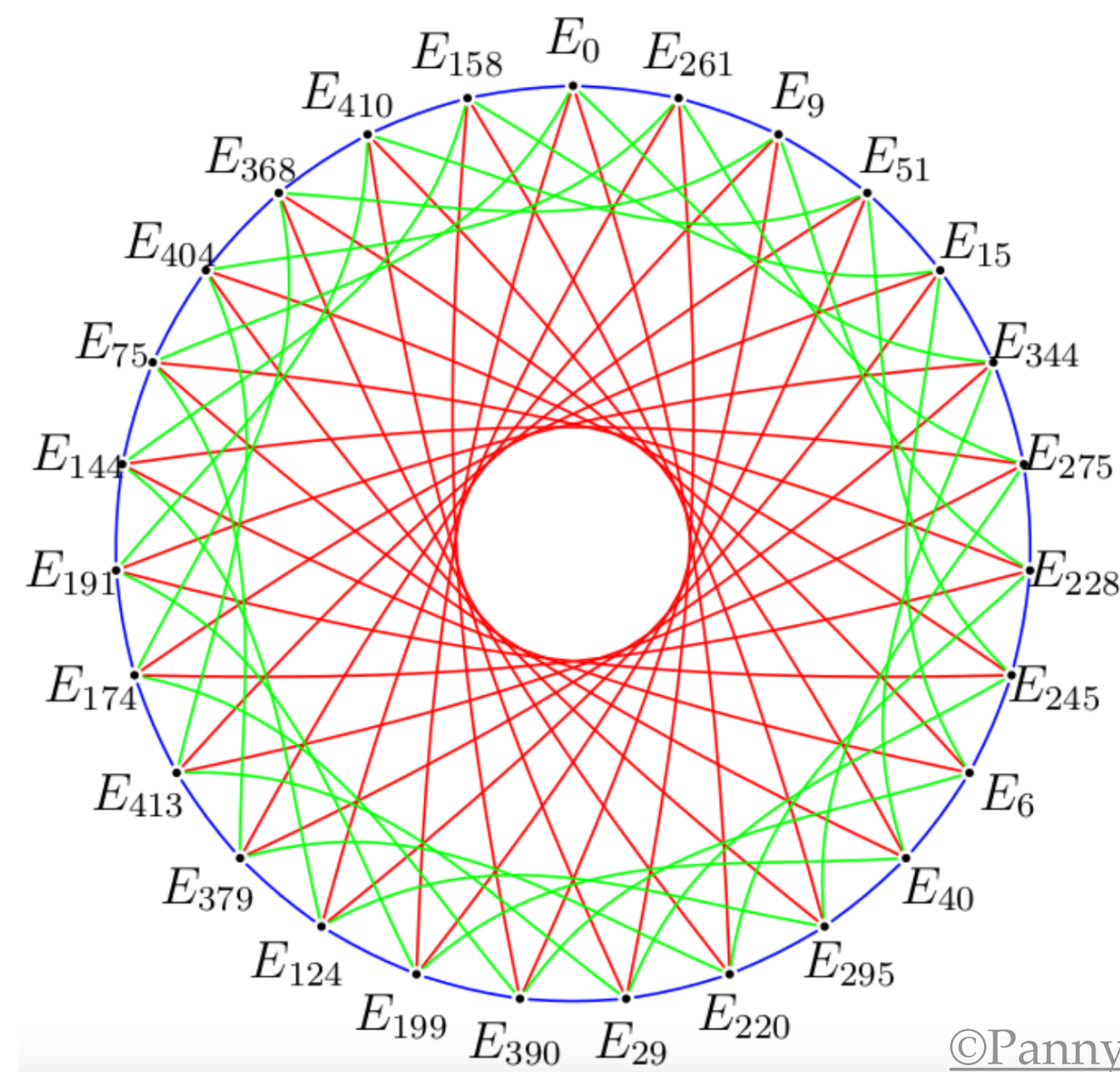
CSIDH

- ▶ Choose small primes ℓ_1, \dots, ℓ_n , making sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime \rightarrow we can compute ℓ_i -steps in the positive or in the negative direction, for all ℓ_i .

Example. CSIDH-512: $p = 4 \cdot \prod \ell_i - 1$, for $\ell_i \in \{3, 5, \dots, 377, 587\}$ (the first 73 primes and 587).

- ▶ Vertices are supersingular curves $y^2 = x^3 + Ax^2 + x$ with $A \in \mathbb{F}_p$.
- ▶ Alice's (Bob's) path is an isogeny of degree $\prod \ell_i^{e_i}$.

Example. CSIDH-512: Exponents are $-5 \leq e_i \leq 5$, for all $1 \leq i \leq 74$.



Example. $p = 419 = 4 \cdot 3 \cdot 5 \cdot 7 - 1$.

Walking the CSIDH graph

Taking a **positive** ℓ_i -step.

- ▶ Find a point $(x, y) \in E$ of **order** ℓ_i with $x, y \in \mathbb{F}_p$.
- ▶ Compute the isogeny with **kernel** $\langle (x, y) \rangle$ using Vélu's formulas.

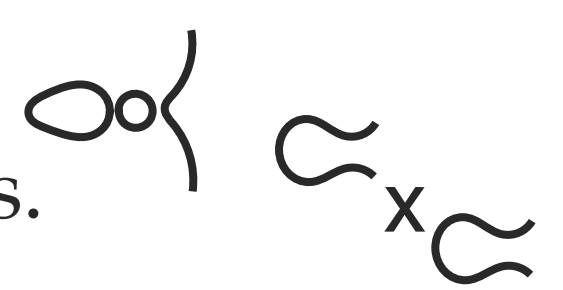
Taking a **negative** ℓ_i -step.

- ▶ Find a point $(x, y) \in E$ of **order** ℓ_i with $x \in \mathbb{F}_{p'}$, but $y \notin \mathbb{F}_p$.
- ▶ Compute the isogeny with **kernel** $\langle (x, y) \rangle$ using Vélu's formulas.

or

- ▶ Go to the quadratic twist. Compute a positive ℓ_i -step. Go to the quadratic twist.

What we did not cover

- ▶ The history of SIDH.
- ▶ Why CSIDH represents an action of an ideal-class group.
- ▶ The Deuring correspondence.
- ▶ Isogenies in higher dimensions. 
- ▶ SQISign (intuition in assignment - then ask me in the next tutorial)
- ▶ Many emerging schemes.
- ▶ ...