Supersingular Isogeny Diffie-Hellman Protocol

Pavel Surý

April 6, 2019

MFF UK

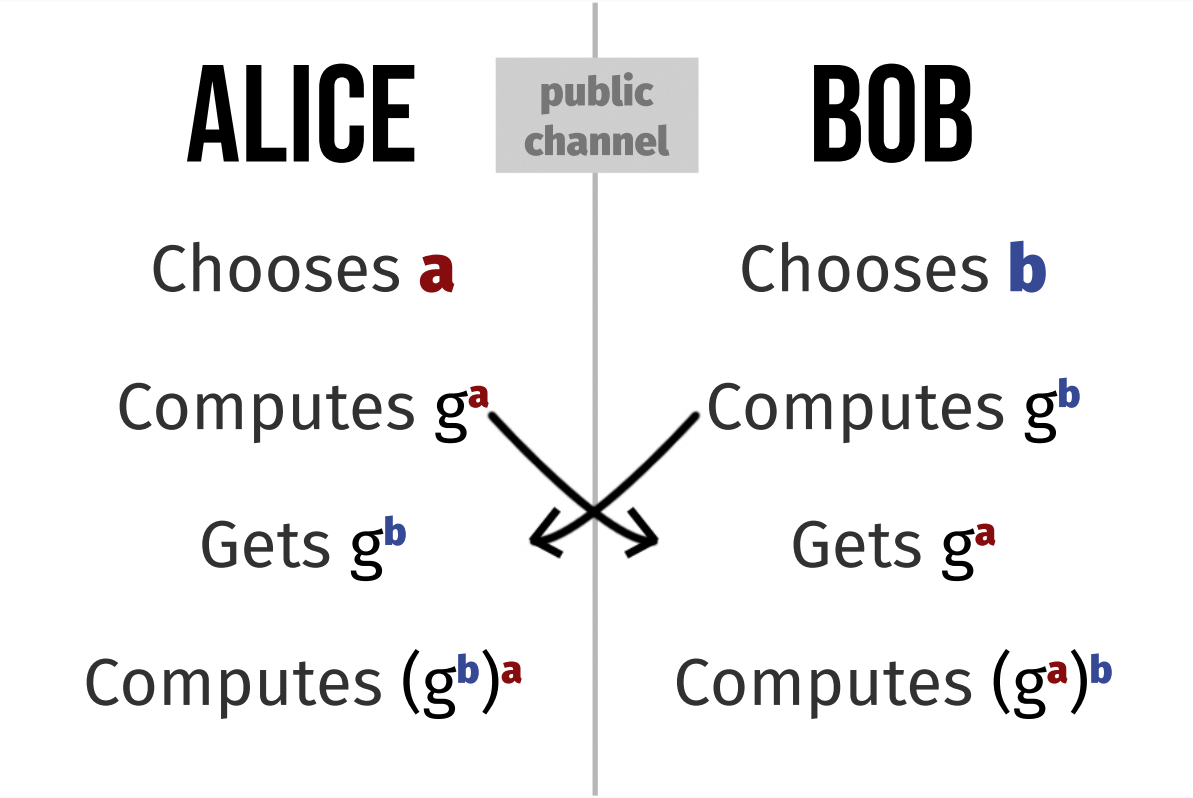
Contents

Contents (Part 1)

1. **Classic Diffie-Hellman** (briefly)
2. Elliptic curves (briefly)
3. Quantum mechanics (briefly)
4. Quantum computation (briefly)
5. Supersingular Isogeny Diffie-Hellman Protocol
   (actually, also briefly)

Diffie-Hellman

Private key exchange over public channel. Let $(G, \cdot, ^{-1}, 1)$ be a group, $g \in G$ fixed.



Attacker sees g^a and g^b. Diffie-Hellman's problem (to obtain g^{ab} from these) is classically hard for some groups G.

Contents (Part 2)

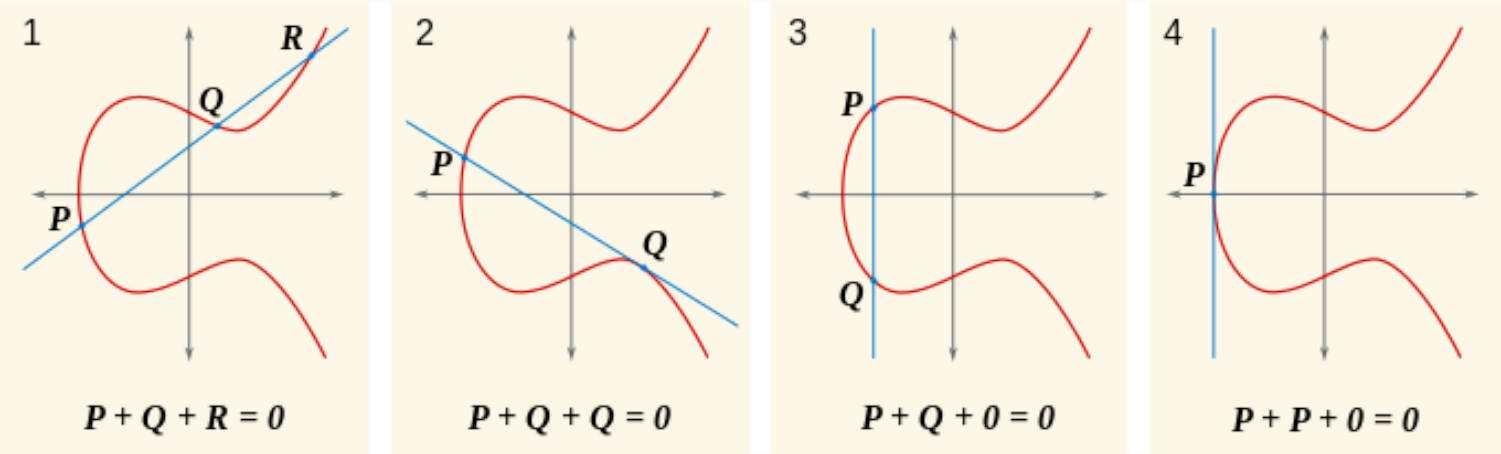
1. Classic Diffie-Hellman (briefly)
2. **Elliptic curves** (briefly)
3. Quantum mechanics (briefly)
4. Quantum computation (briefly)
5. Supersingular Isogeny Diffie-Hellman Protocol
   (actually, also briefly)

Elliptic curves

Elliptic curve is a **non-singular** curve over finite field $\mathbb{F}$ defined by equation

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

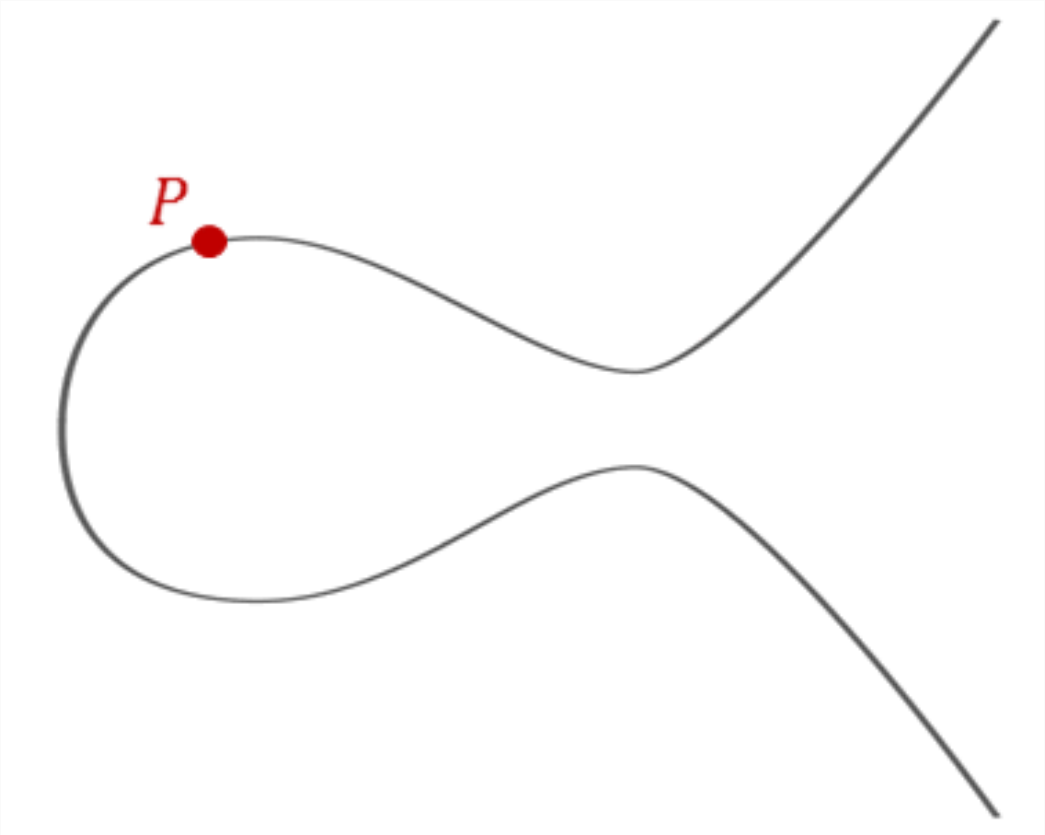
It is possible to define a group structure on the set of solutions.



We also need to consider the ∞ point on the curve so that every line intersects the curve in exactly three points including multiplicity (algebraic geometry).

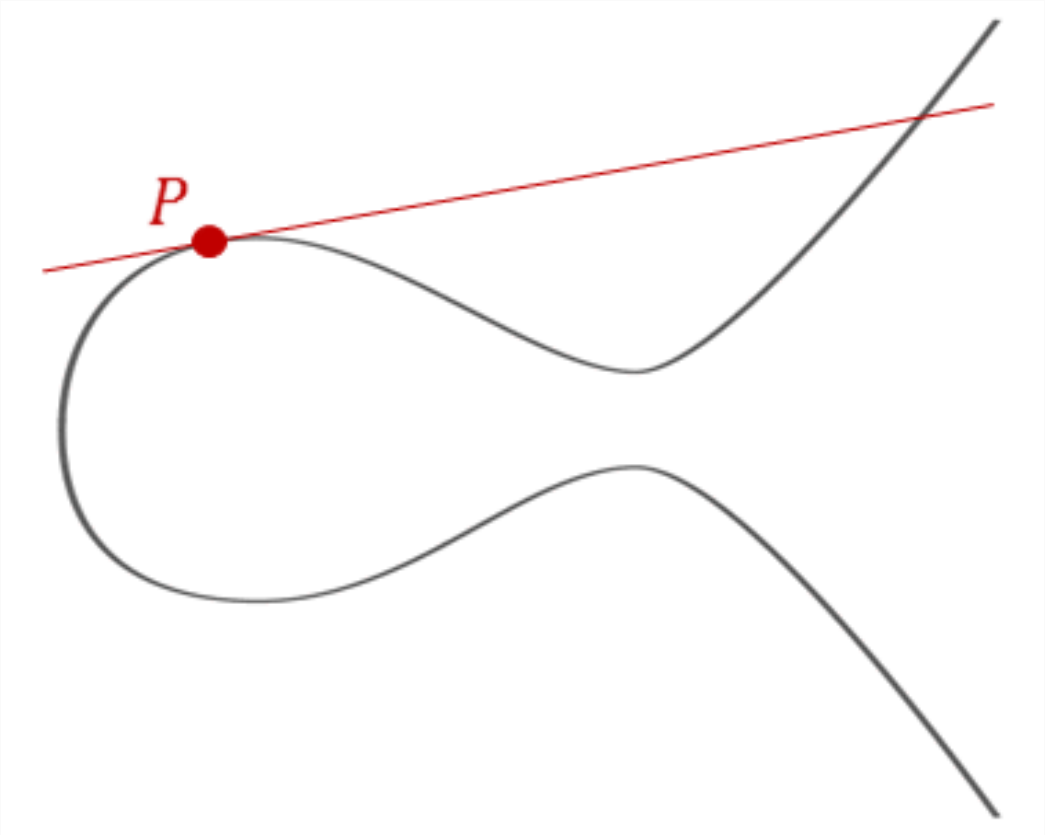
Elliptic curves

If we represent the group of elliptic curves additively:



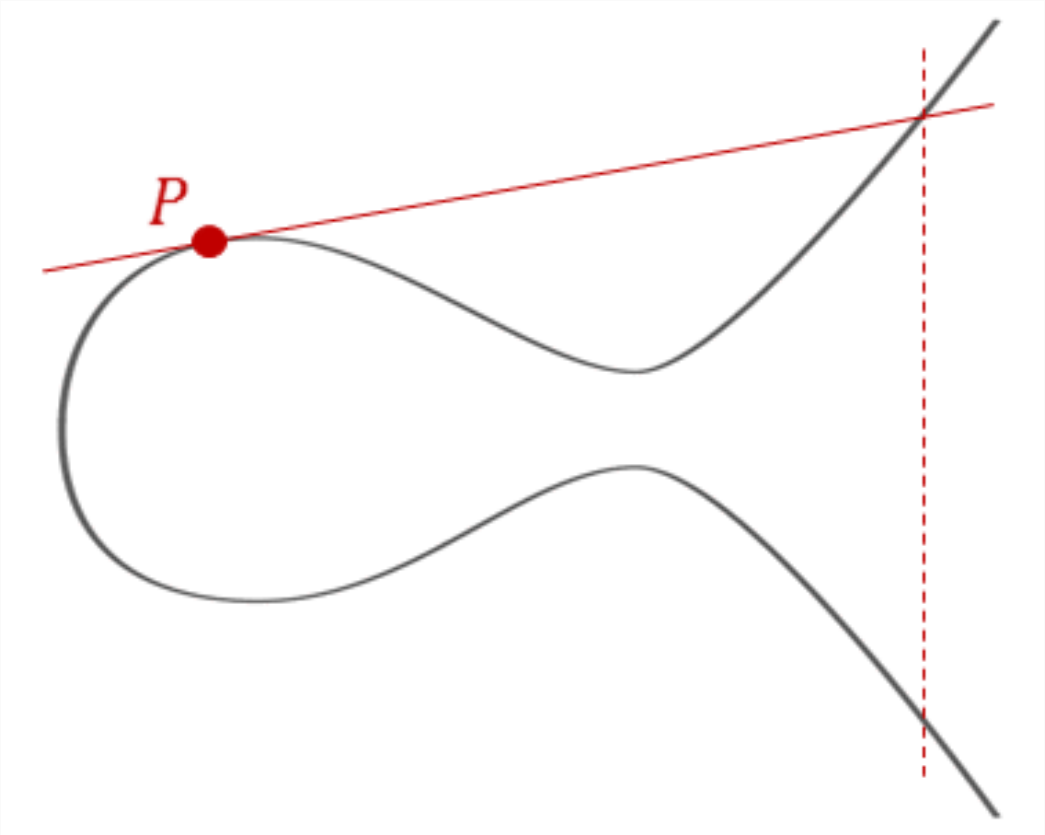
Elliptic curves

If we represent the group of elliptic curves additively:



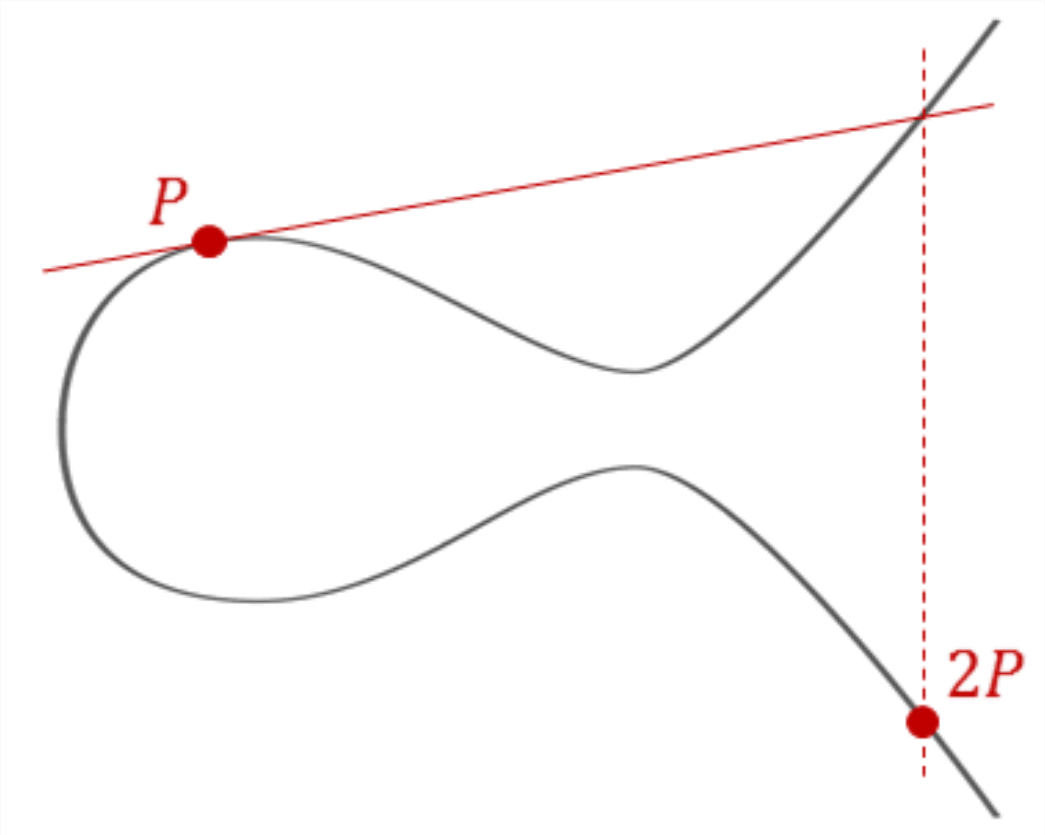
Elliptic curves

If we represent the group of elliptic curves additively:



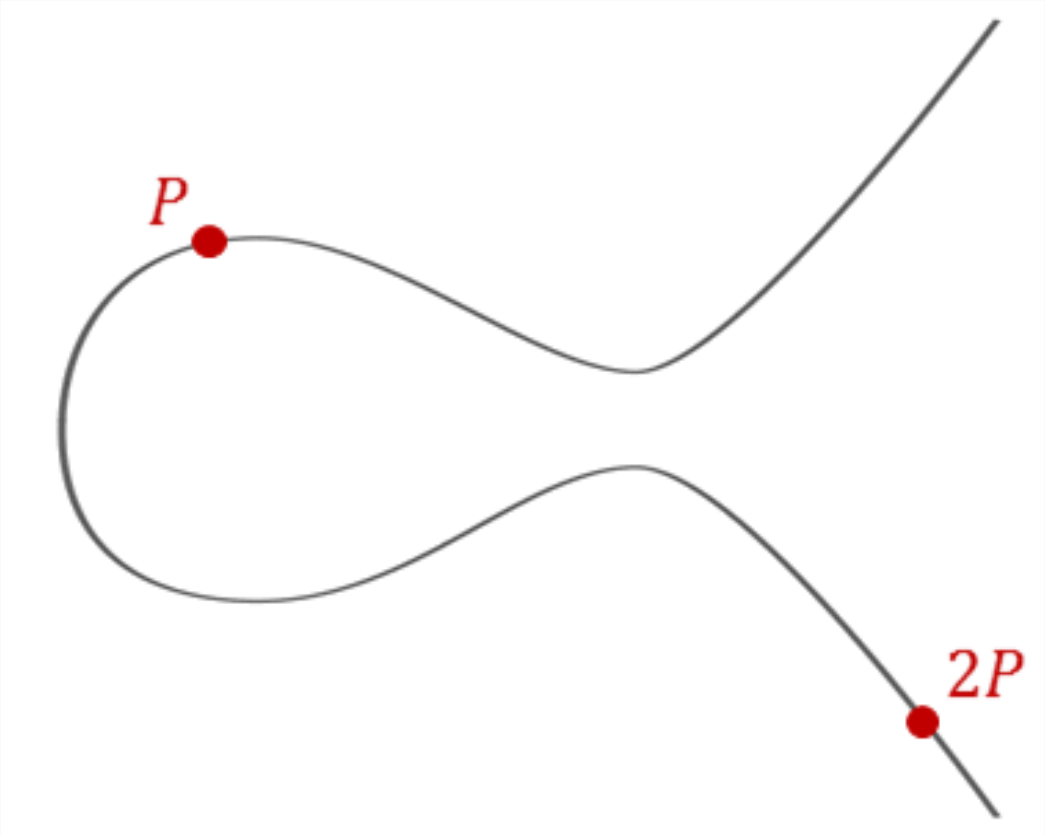
Elliptic curves

If we represent the group of elliptic curves additively:



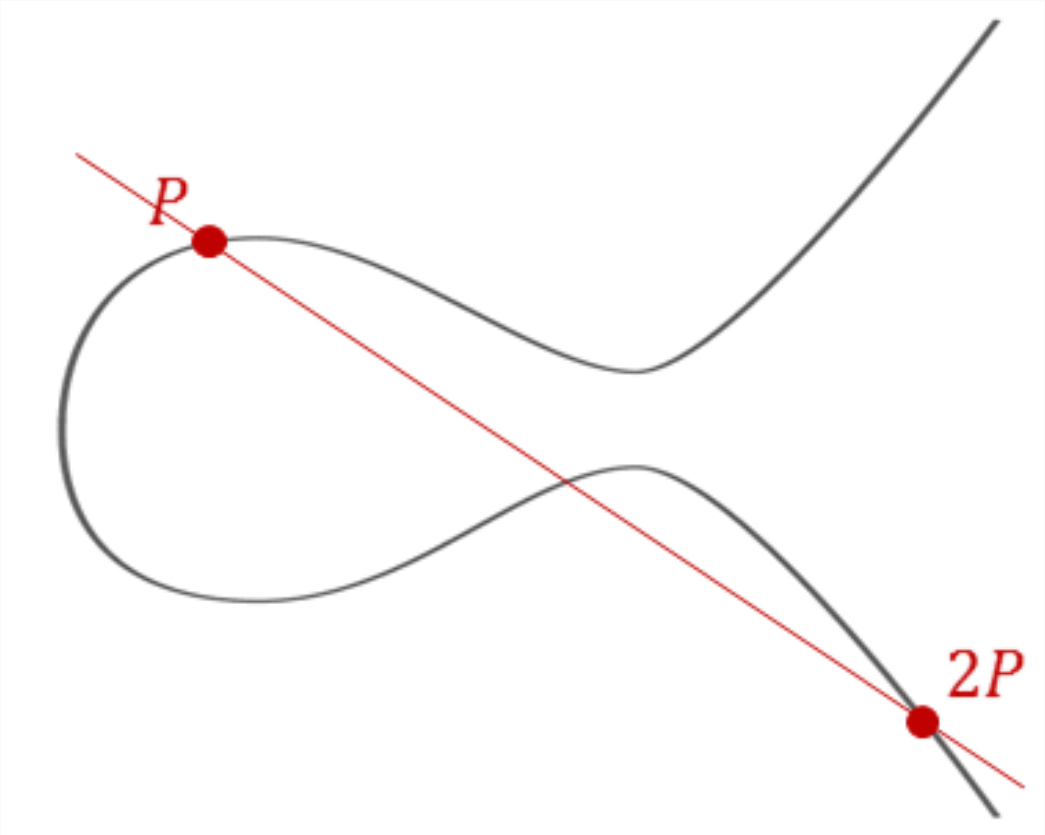
Elliptic curves

If we represent the group of elliptic curves additively:



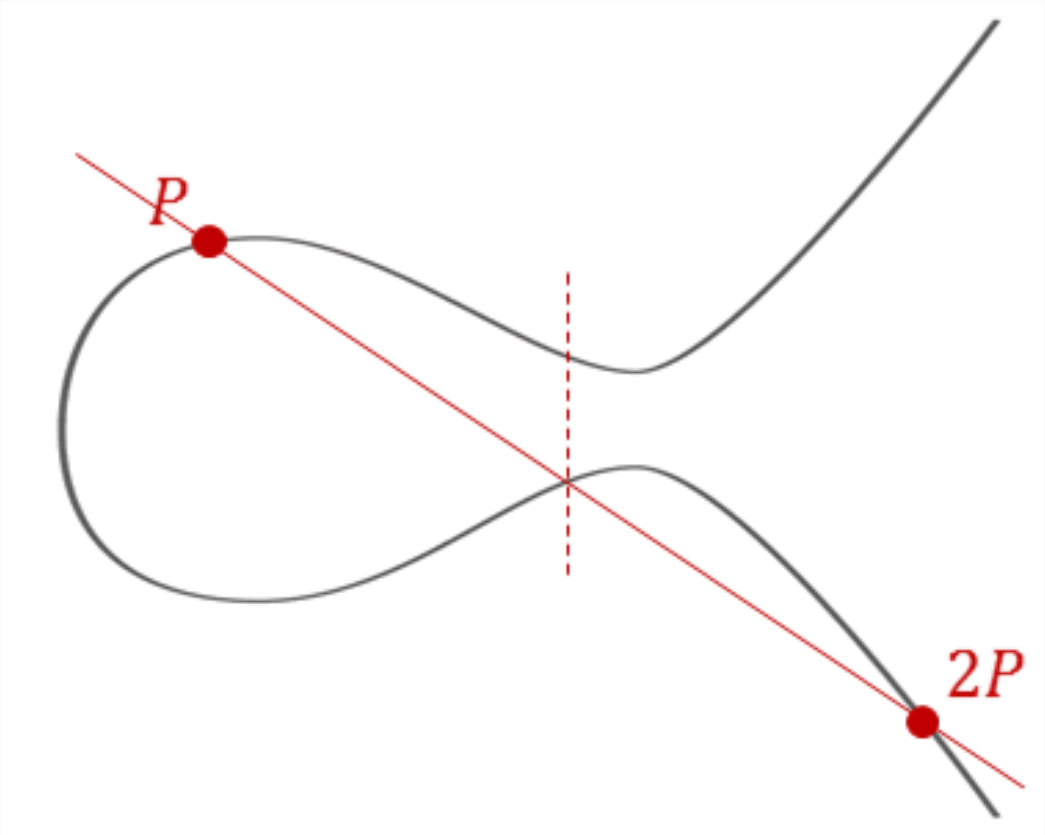
Elliptic curves

If we represent the group of elliptic curves additively:



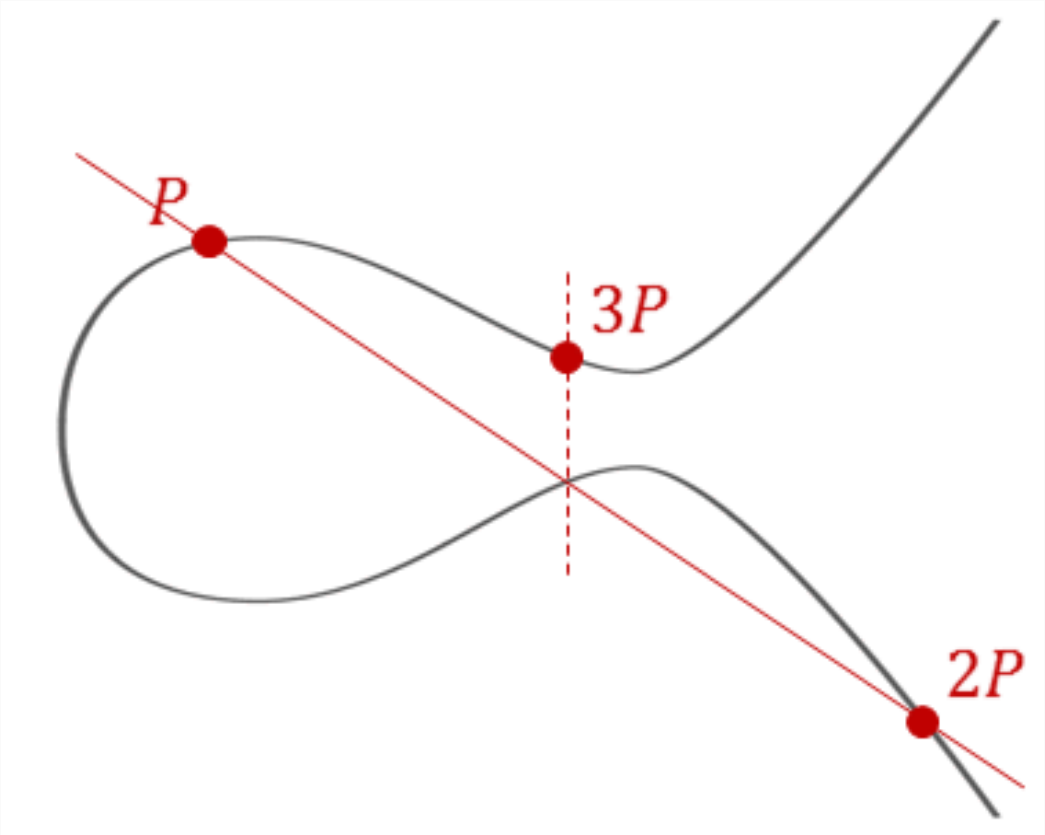
Elliptic curves

If we represent the group of elliptic curves additively:



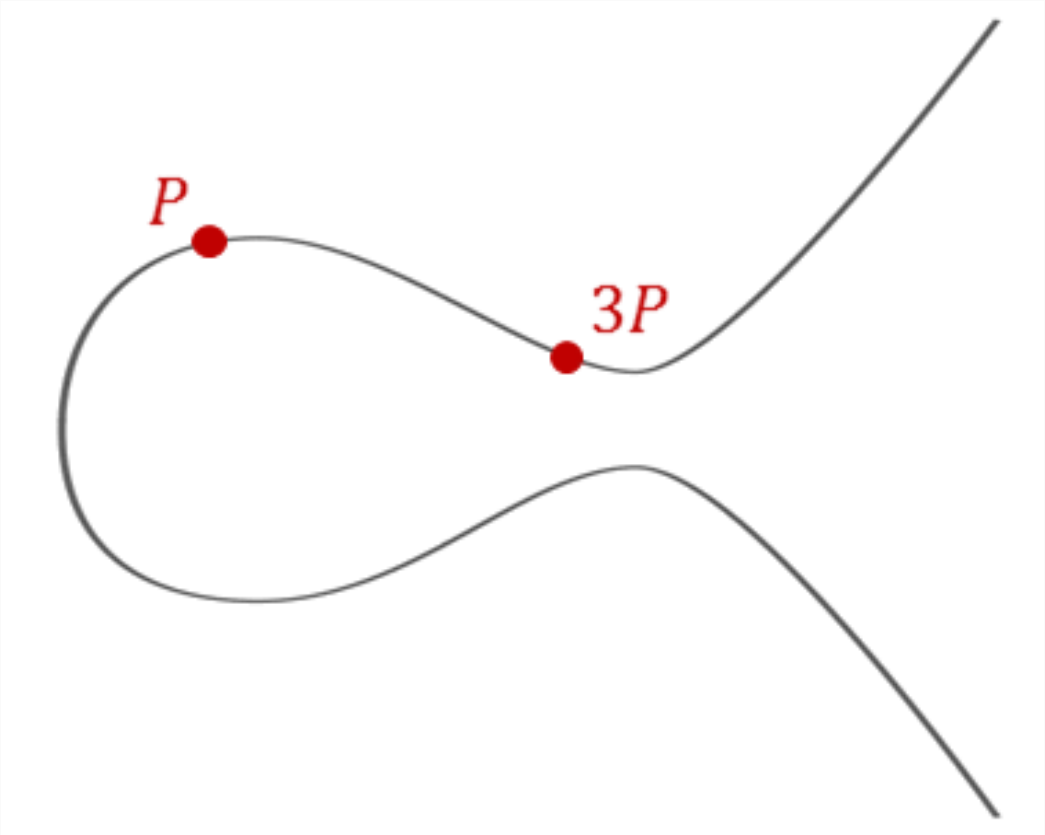
Elliptic curves

If we represent the group of elliptic curves additively:



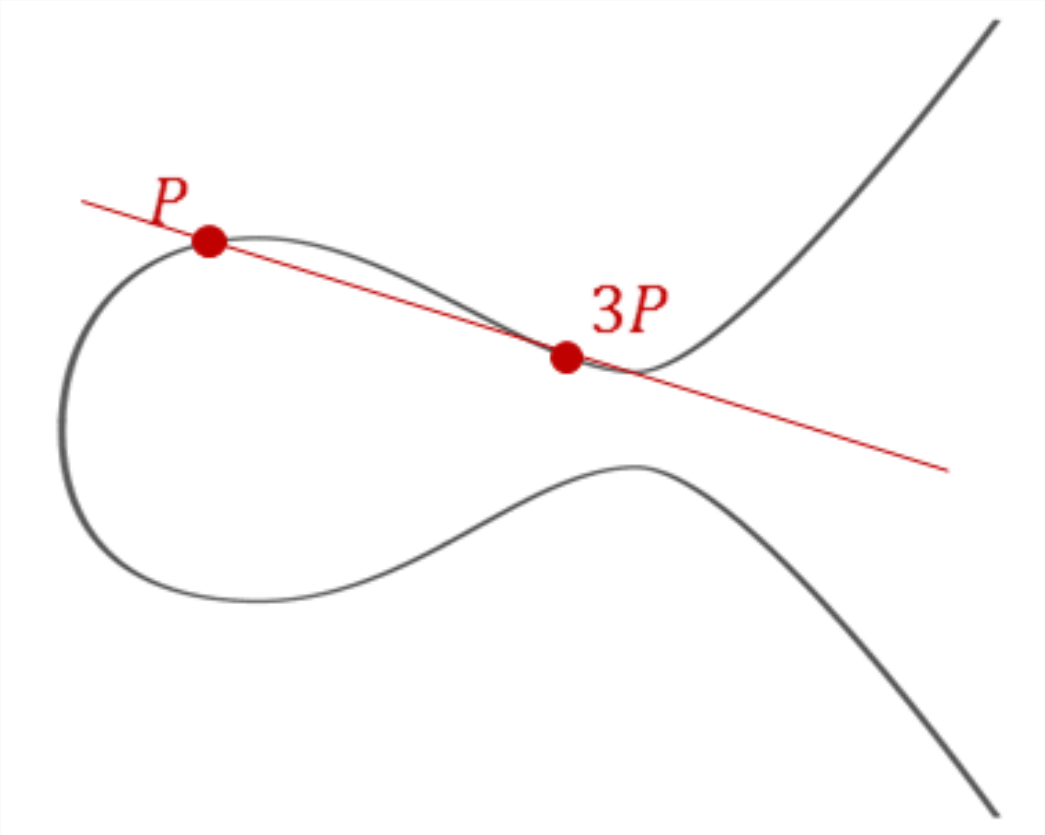
Elliptic curves

If we represent the group of elliptic curves additively:



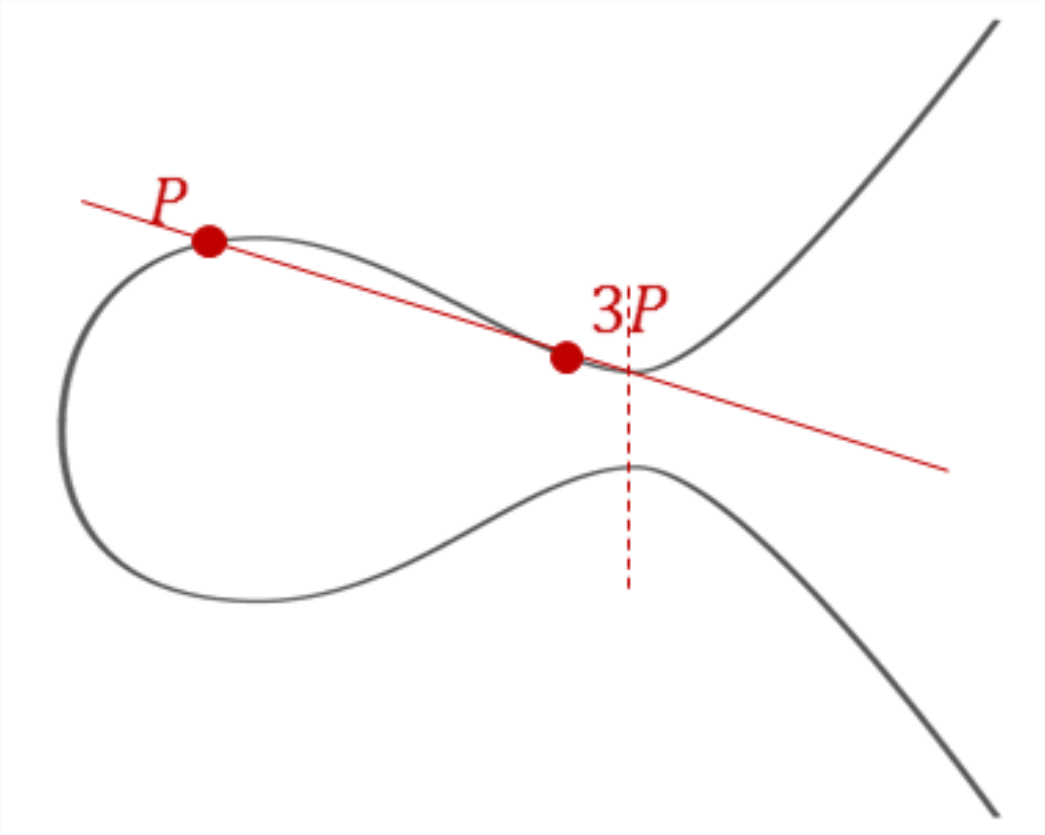
Elliptic curves

If we represent the group of elliptic curves additively:



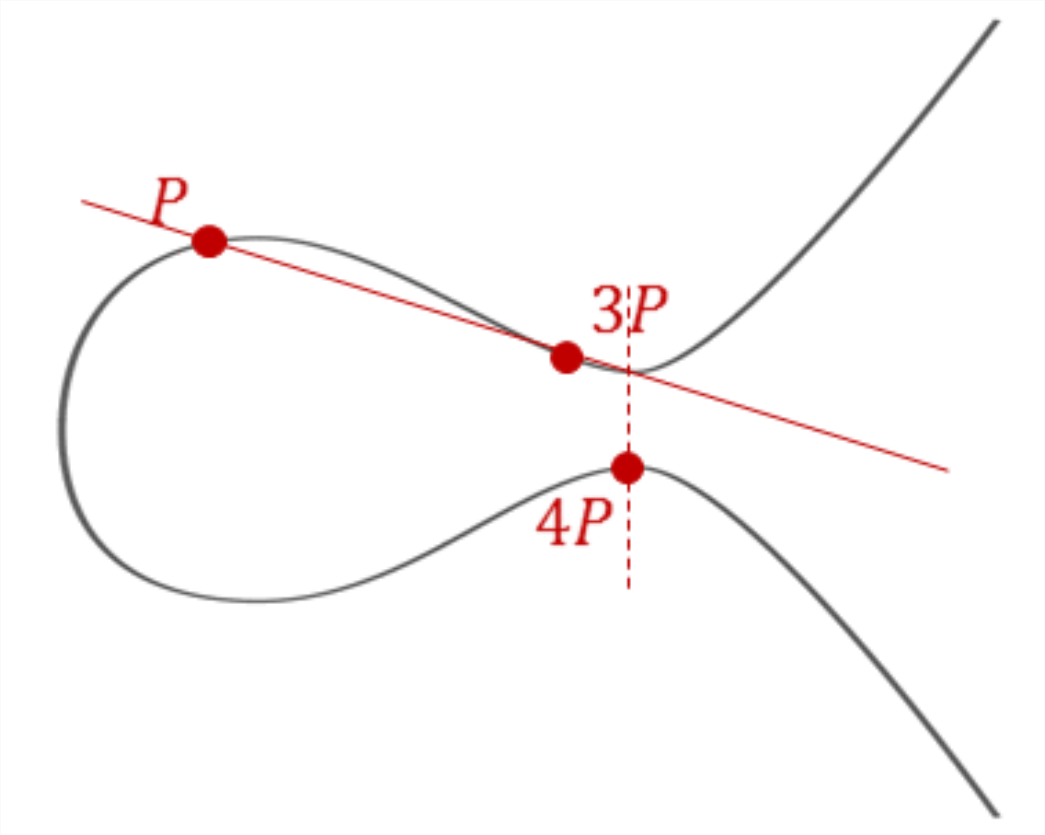
Elliptic curves

If we represent the group of elliptic curves additively:



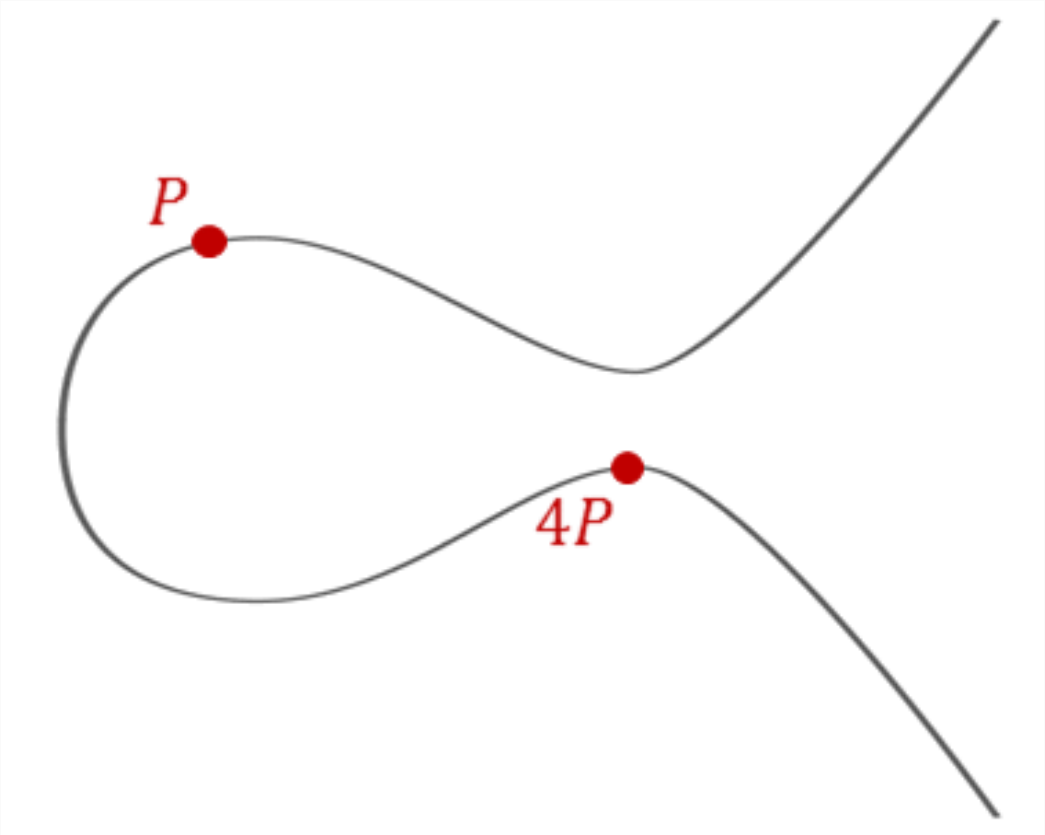
Elliptic curves

If we represent the group of elliptic curves additively:



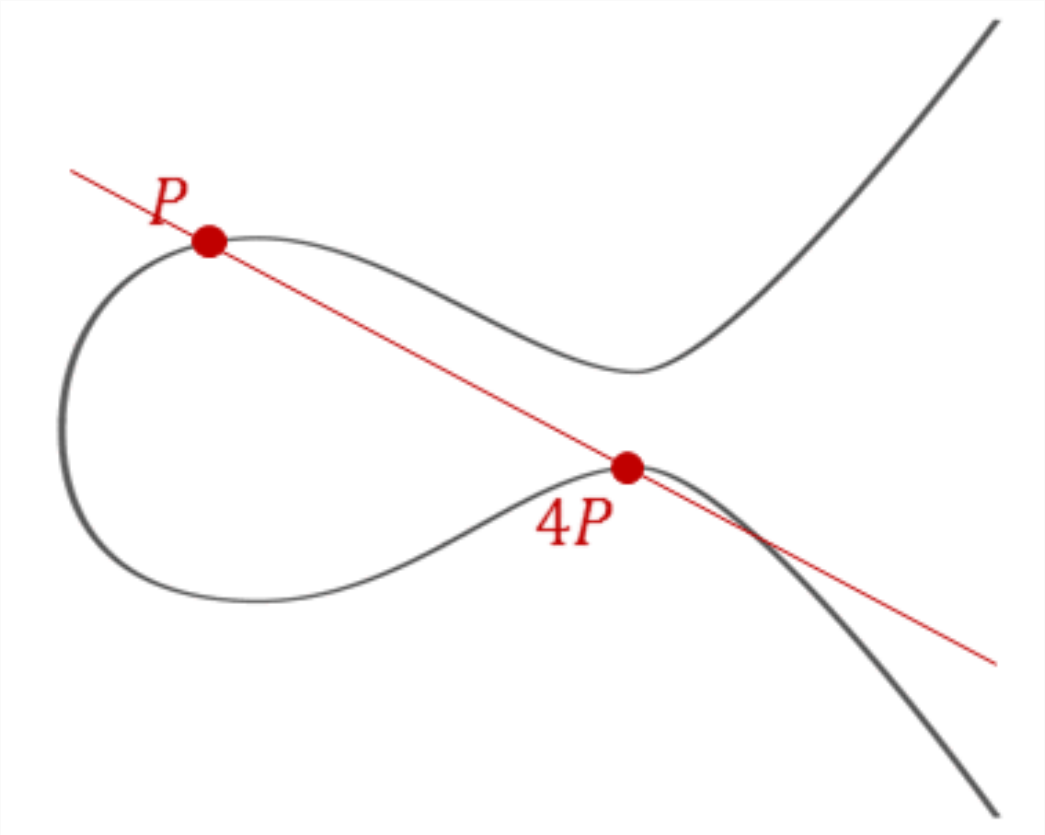
Elliptic curves

If we represent the group of elliptic curves additively:



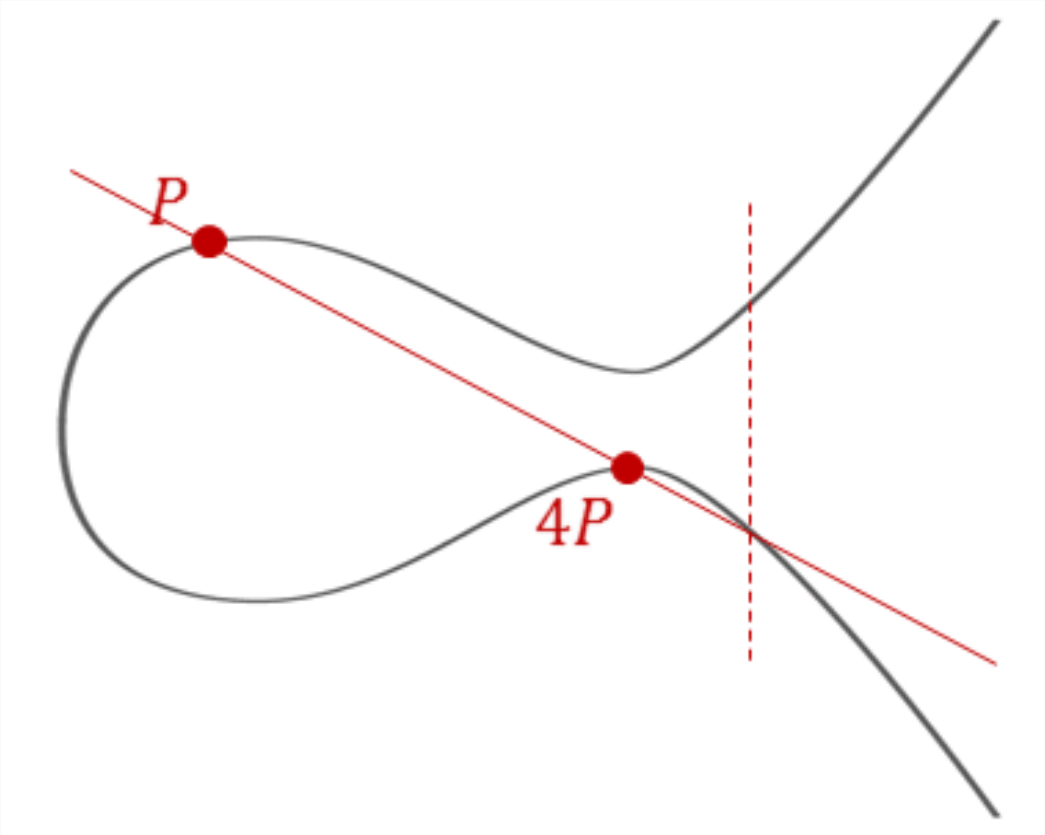
Elliptic curves

If we represent the group of elliptic curves additively:



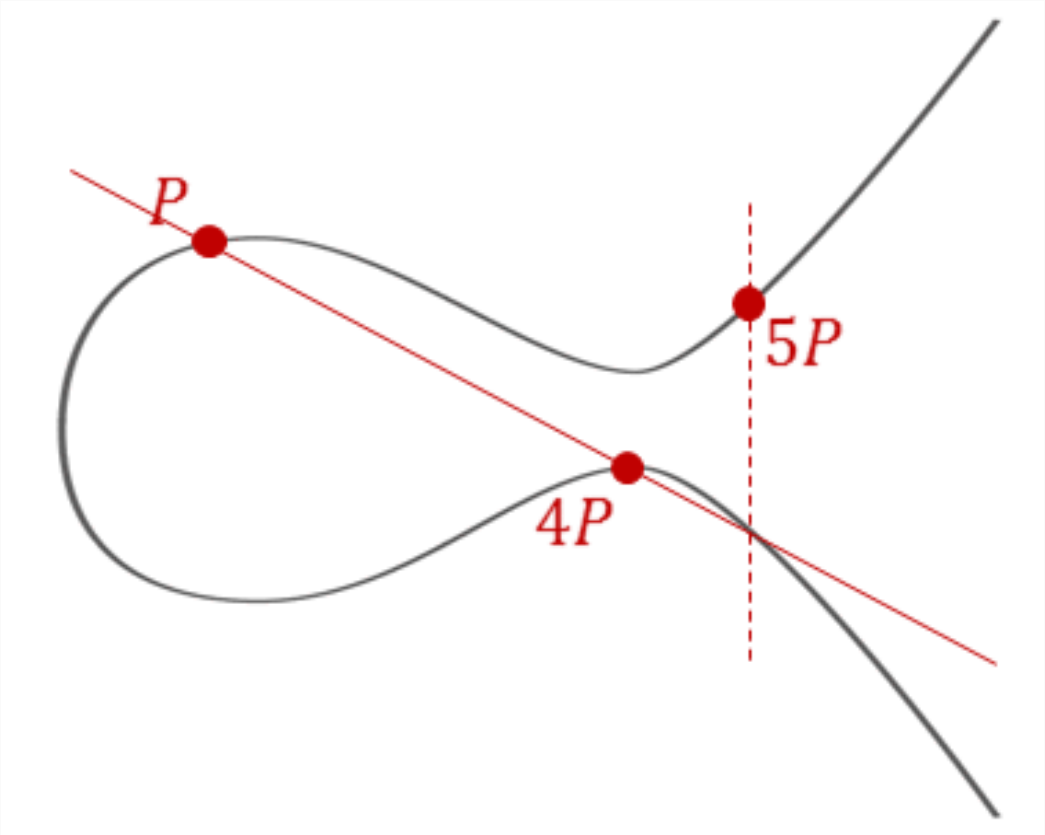
Elliptic curves

If we represent the group of elliptic curves additively:



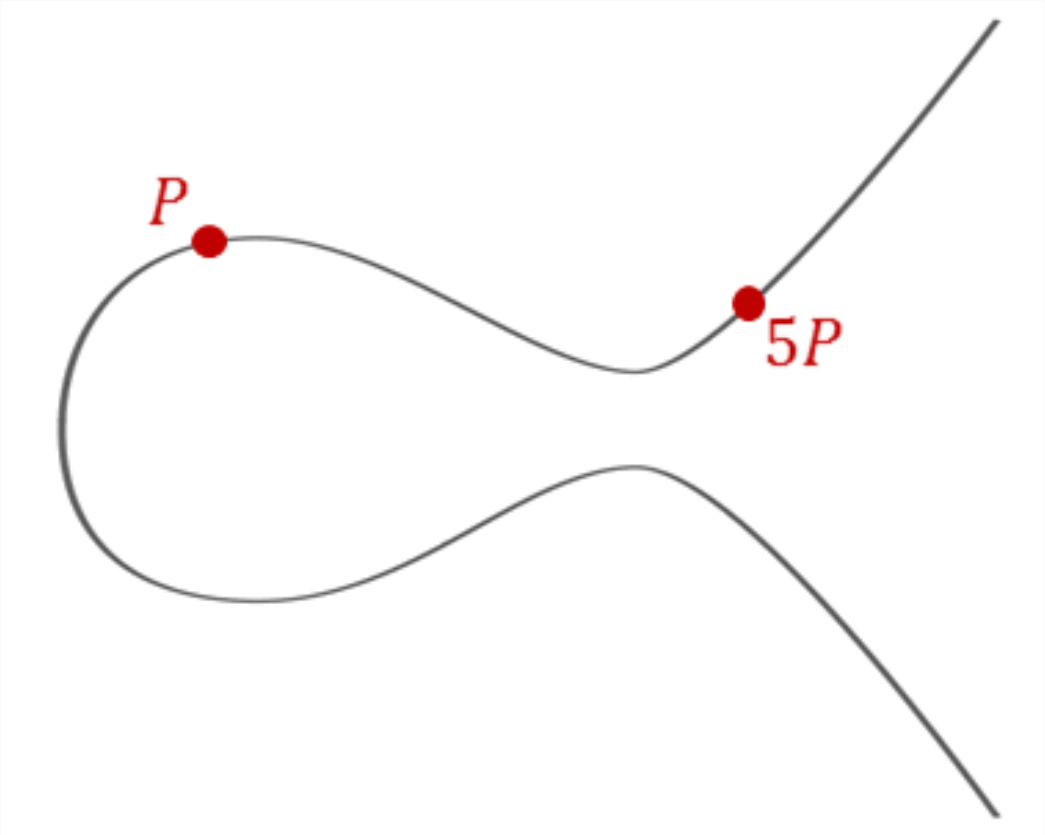
Elliptic curves

If we represent the group of elliptic curves additively:



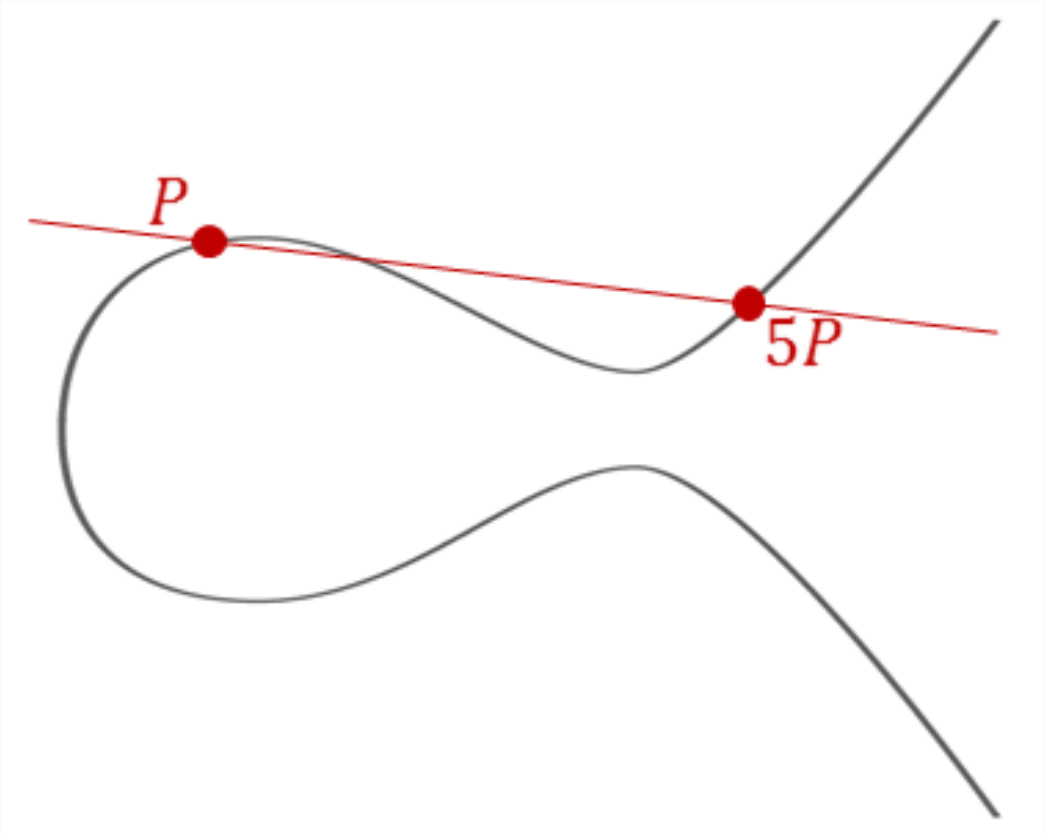
Elliptic curves

If we represent the group of elliptic curves additively:



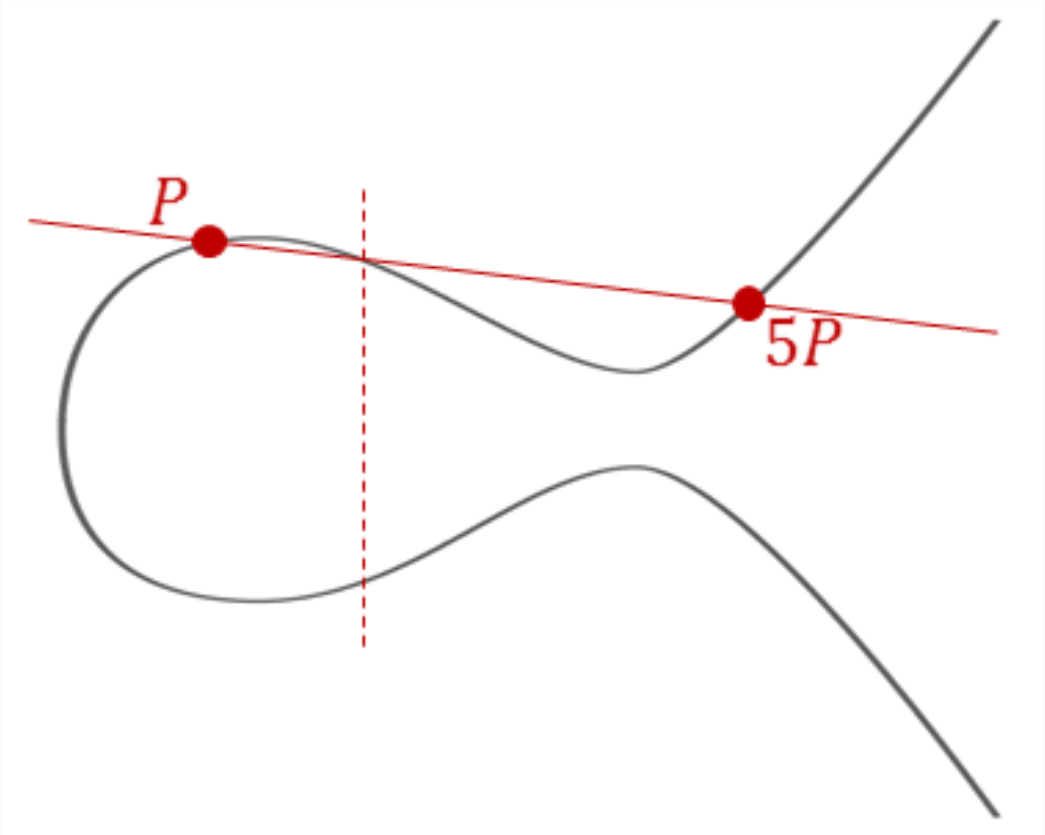
Elliptic curves

If we represent the group of elliptic curves additively:



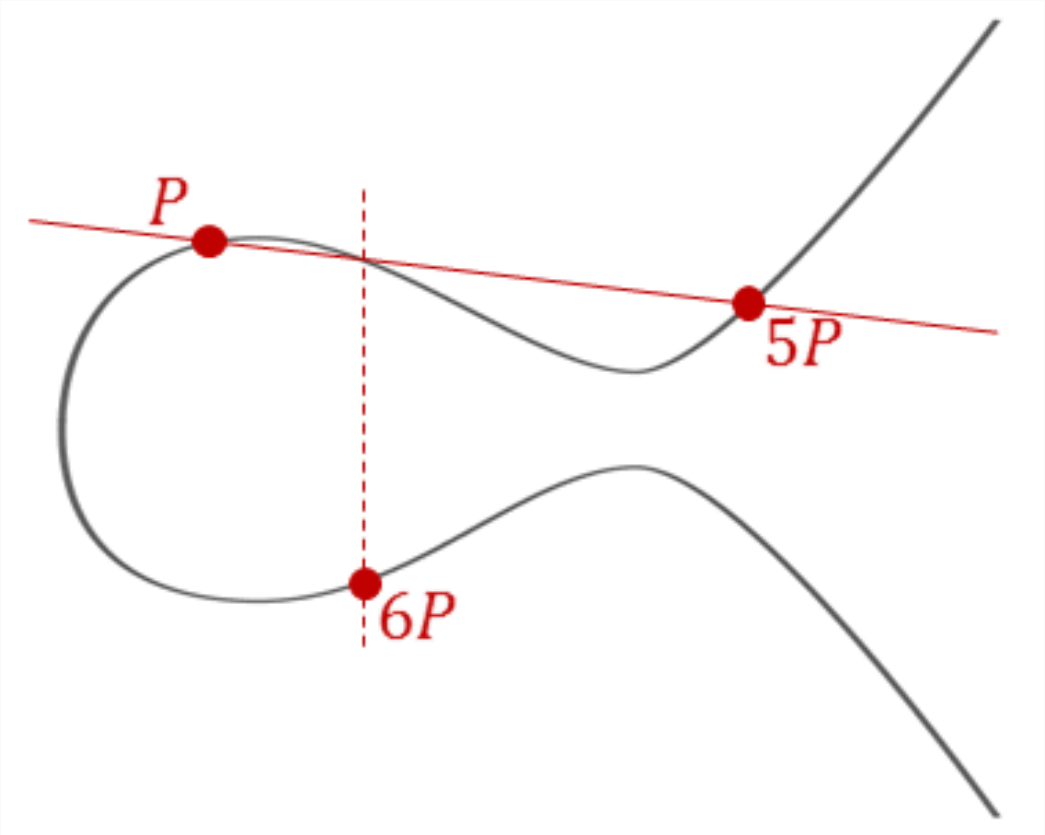
Elliptic curves

If we represent the group of elliptic curves additively:



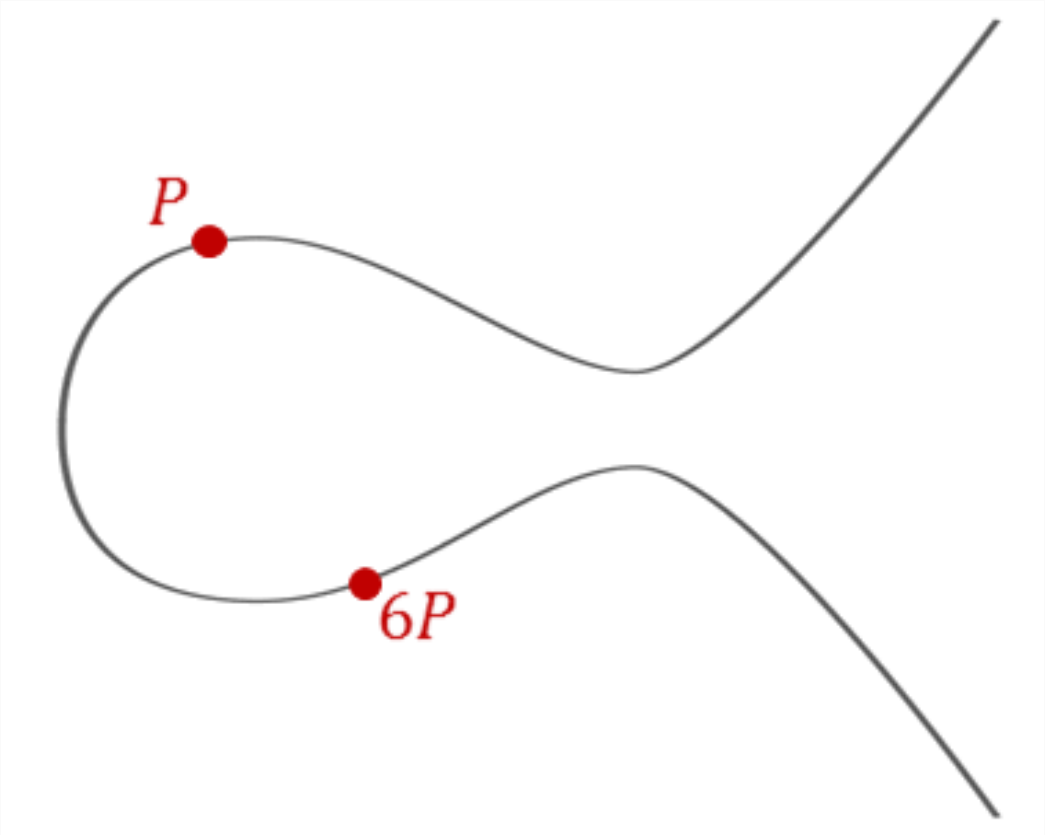
Elliptic curves

If we represent the group of elliptic curves additively:



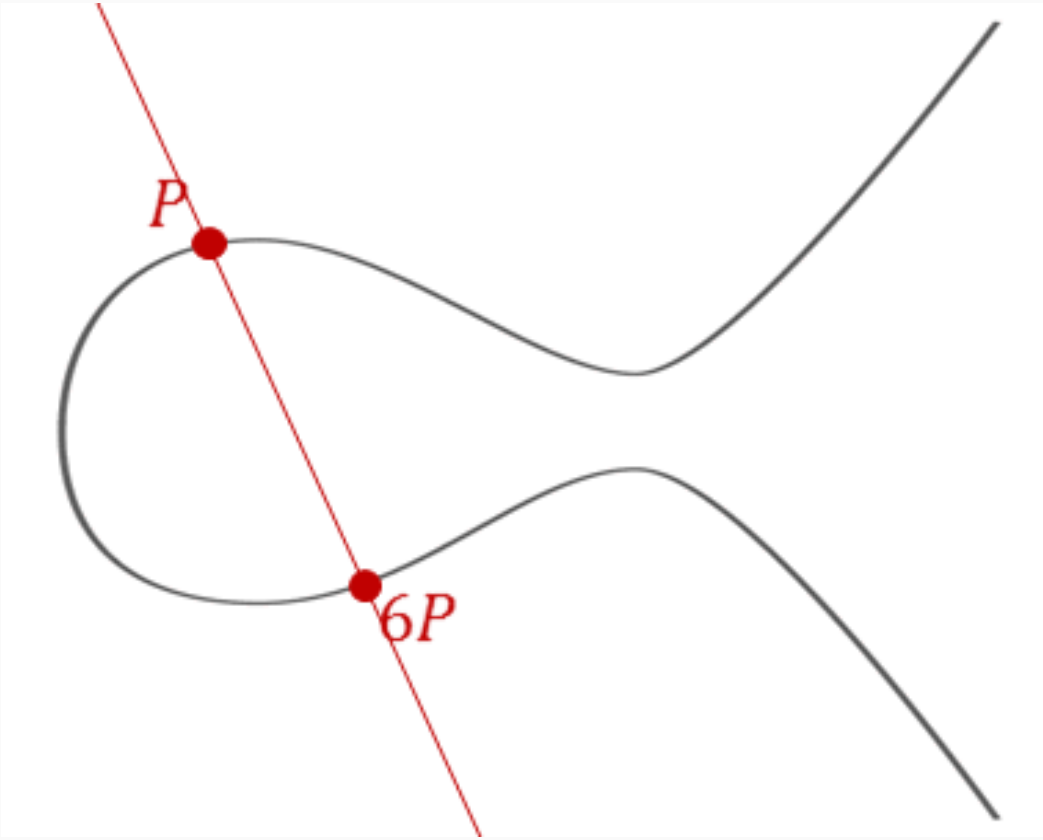
Elliptic curves

If we represent the group of elliptic curves additively:



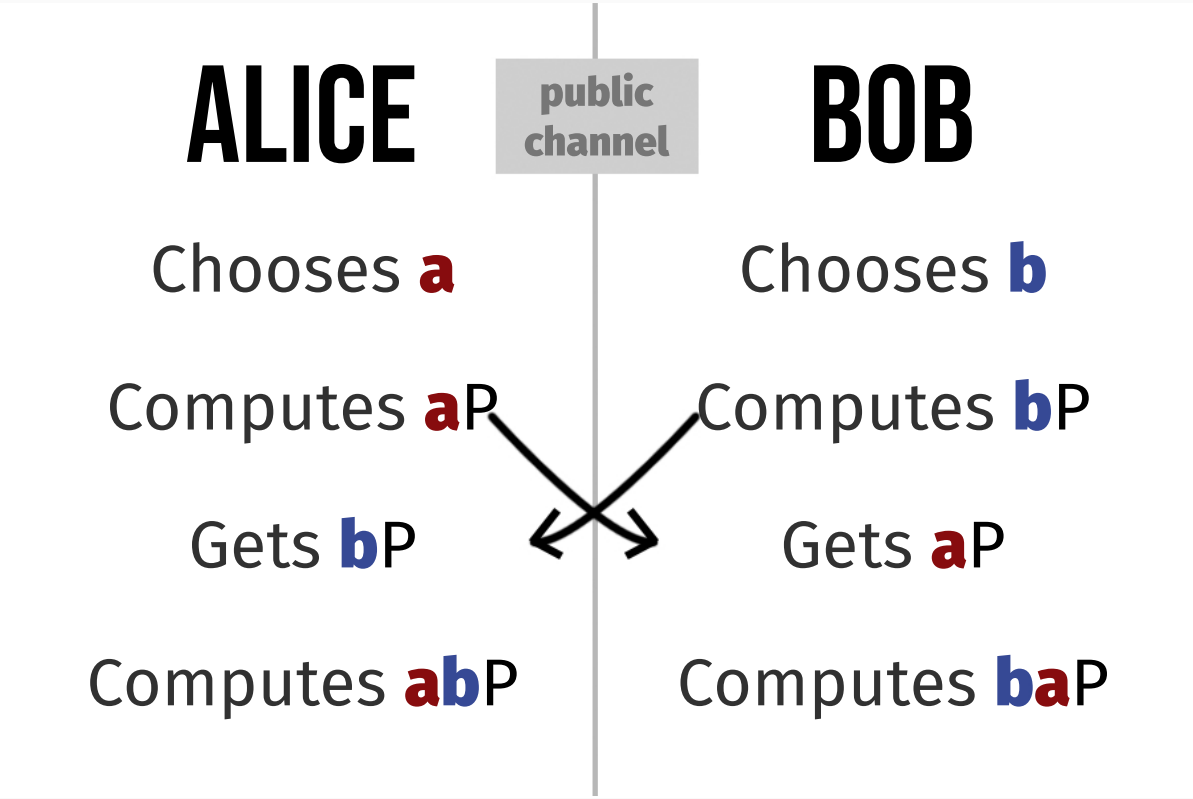
Elliptic curves

If we represent the group of elliptic curves additively:



Elliptic curves

Let $(E, +, -, 0)$ be the group of elliptic curves and P a fixed point.



This way, Alice and Bob can publicly choose a secret point on the curve.

Contents (part 3)

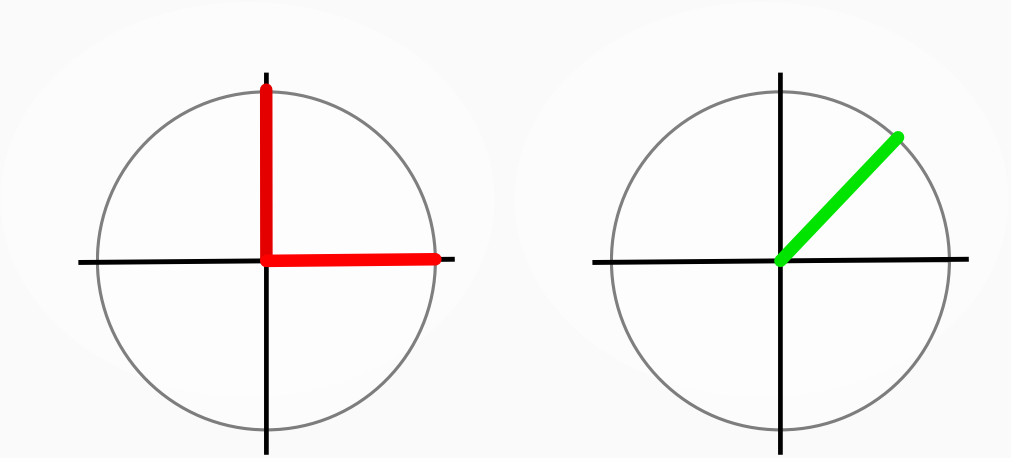
1. Classic Diffie-Hellman (briefly)
2. Elliptic curves (briefly)
3. **Quantum mechanics** (briefly)
4. Quantum computation (briefly)
5. Supersingular Isogeny Diffie-Hellman Protocol
   (actually, also briefly)

Quantum mechanics

Particles are in a superposition of states.

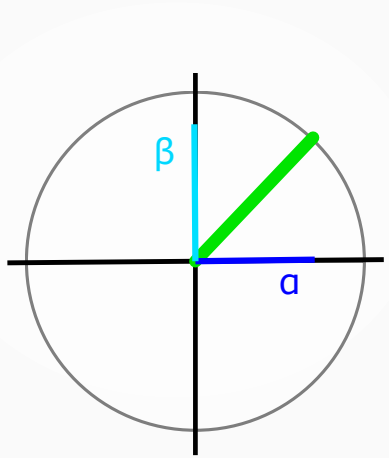
- Bit: $\mathbf{0}$ or $\mathbf{1}$.
- Quantum bit: $\alpha \cdot \mathbf{0} + \beta \cdot \mathbf{1}$, where $\alpha, \beta \in \mathbb{C}$.

We will require that qubits have norm 1, so $\alpha^2 + \beta^2 = 1$.



Quantum mechanics

When measured, the bit collapses to **0** with probability α^2, and to **1** with probability β^2.



Everything works linearly, transitions between the states are unitary.

Contents (part 4)

1. Classic Diffie-Hellman (briefly)
2. Elliptic curves (briefly)
3. Quantum mechanics (briefly)
4. **Quantum computation** (briefly)
5. Supersingular Isogeny Diffie-Hellman Protocol
   (actually, also briefly)

Quantum computation

Working on superpositions can be beneficial for computation.

- can do the Fourier transform quickly,

Quantum computation

Working on superpositions can be beneficial for computation.

- can do the Fourier transform quickly,
- so they can find the periods of functions quickly,

Quantum computation

Working on superpositions can be beneficial for computation.

- can do the Fourier transform quickly,
- so they can find the periods of functions quickly,
- so they can decide the orders of elements in groups,

Quantum computation

Working on superpositions can be beneficial for computation.

- can do the Fourier transform quickly,
- so they can find the periods of functions quickly,
- so they can decide the orders of elements in groups,
- so they can break RSA / ECC / standard Diffie-Hellman.

Quantum computation

- 2330-qubit computer could break a 256-bit elliptic curve,

Quantum computation

- 2330-qubit computer could break a 256-bit elliptic curve,
- 4098-qubit computer could break 2048-bit RSA key,

Quantum computation

- 2330-qubit computer could break a 256-bit elliptic curve,
- 4098-qubit computer could break 2048-bit RSA key,
- but in practice, we have only 10-qubit computers so far.

Quantum computation

The biggest problem in quantum computation is noise. Two possible scenarios:

- Optimistic: We can realize any quantum computation with bounded error probability (independent on number of qubits).
- Pessimistic: Computation error (noise) grows with the number of qubits at least linearly.

Post-quantum cryptography

- Algorithms resistant to known quantum attacks started to emerge.
- American NIST called for proposals (deadline of November 30, 2017).

Contents (part 5)

1. Classic Diffie-Hellman (briefly)
2. Elliptic curves (briefly)
3. Quantum mechanics (briefly)
4. Quantum computation (briefly)
5. **Supersingular Isogeny Diffie-Hellman Protocol**
   (actually, also briefly)

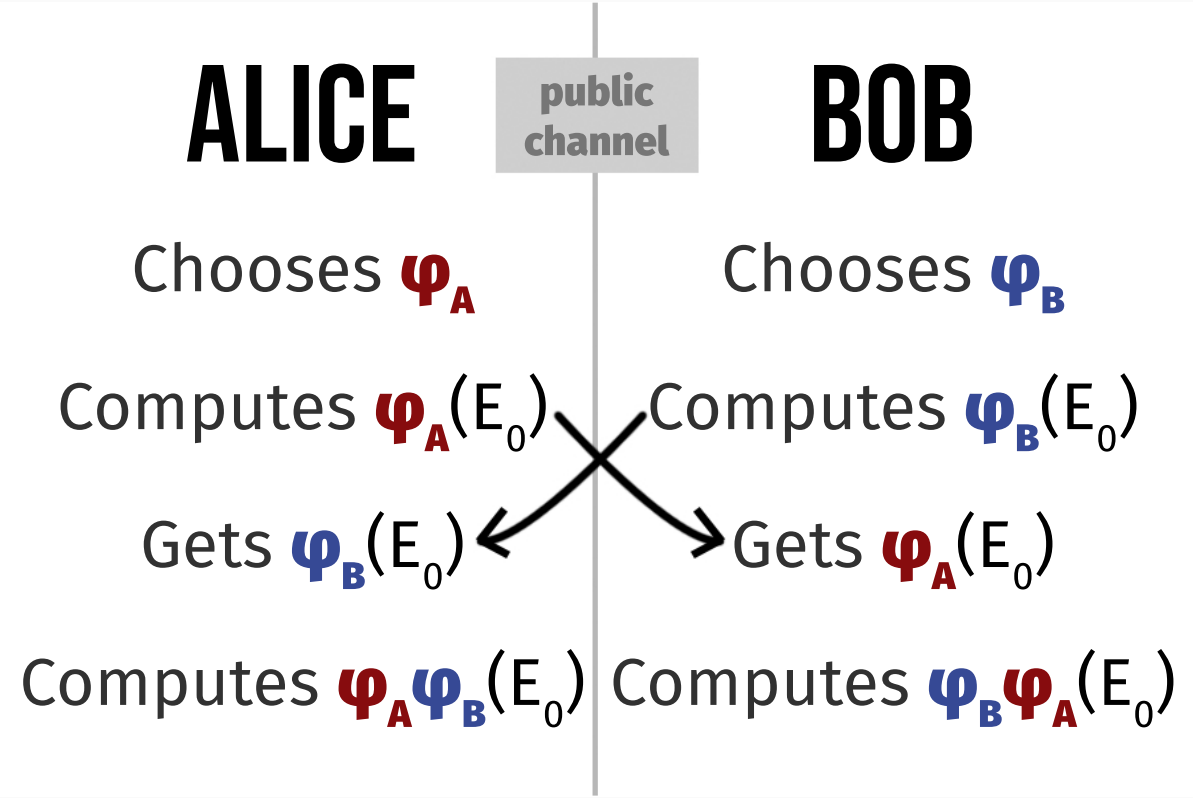
Supersingular Isogeny Diffie-Hellman

Uses not only one elliptic curve, but multiple ones - it uses a walk on supersingular isogeny graph.

Underlying problem is not discrete logarithm, but finding a connection between two elliptic curves E and E'.

Supersingular Isogeny Diffie-Hellman

The Diffie-Hellman would end up like this:



We will construct something similar, yet a bit different.

Supersingular elliptic curves

For security reasons, we need to use supersingular elliptic curves (for normal ones, the underlying problem is easier).

Definition
Elliptic curve E is supersingular, if the endomorphism ring $End(E)$ has rank 4. (normal elliptic curves have endomorphism ring of rank 1 or 2).

Definition
j-invariant of an elliptic curve is $j(E) = 1728\frac{4a^3}{4a^3+27b^2}$.

When $j(E) \in \{0, 1728\}$, E is supersingular. This happens iff $a = 0$ or $b = 0$.

Isogenies

Isogenies are morphisms between elliptic curves $\varphi : E_1 \to E_2$ such that the point at infinity in E_1 is mapped to the point at infinity in E_2.

Isogeny is uniquely determined by $Ker(\varphi)$. When we know E_1 and $Ker(\varphi)$, there is a set of formulas (**Velu's formulas**) that determine $\varphi(x)$ for any $x \in E_1$.

Algorithm (idea)

1. Alice and Bob agree on a pool of curves and a starting curve E_0.
2. Alice applies a secret random sequence of two-isogenies, resulting in an elliptic curve $E_A = \varphi_A(E_0)$.
3. Bob secretly walks a random trail of three-isogenies in order to end up at a curve $E_B = \varphi_B(E_0)$.
4. They exchange curves E_A and E_B, and reapply the isogenies on the received curve.
5. They both end up with the same curve, E_{AB}.

(gif)

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.
2. Alice calculates a point $R_A = (m_A)P_A + (n_A)Q_A$.

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.
2. Alice calculates a point $R_A = (m_A)P_A + (n_A)Q_A$.
3. This uniquely determines isogeny φ_A such that $Ker(\varphi_A) = R_A$.

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.
2. Alice calculates a point $R_A = (m_A)P_A + (n_A)Q_A$.
3. This uniquely determines isogeny φ_A such that $Ker(\varphi_A) = R_A$.
4. Alice sends E_A, $\varphi_A(P_B)$ and $\varphi_A(Q_B)$ to Bob.

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.
2. Alice calculates a point $R_A = (m_A)P_A + (n_A)Q_A$.
3. This uniquely determines isogeny φ_A such that $Ker(\varphi_A) = R_A$.
4. Alice sends E_A, $\varphi_A(P_B)$ and $\varphi_A(Q_B)$ to Bob.
5. Alice receives E_B, $\varphi_B(P_A)$ and $\varphi_B(Q_A)$.

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.
2. Alice calculates a point $R_A = (m_A)P_A + (n_A)Q_A$.
3. This uniquely determines isogeny φ_A such that $Ker(\varphi_A) = R_A$.
4. Alice sends E_A, $\varphi_A(P_B)$ and $\varphi_A(Q_B)$ to Bob.
5. Alice receives E_B, $\varphi_B(P_A)$ and $\varphi_B(Q_A)$.
6. Alice calculates φ'_A as isogeny with kernel $(m_A)\varphi_B(P_A) + (n_A)\varphi_B(Q_A)$.

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.
2. Alice calculates a point $R_A = (m_A)P_A + (n_A)Q_A$.
3. This uniquely determines isogeny φ_A such that $Ker(\varphi_A) = R_A$.
4. Alice sends E_A, $\varphi_A(P_B)$ and $\varphi_A(Q_B)$ to Bob.
5. Alice receives E_B, $\varphi_B(P_A)$ and $\varphi_B(Q_A)$.
6. Alice calculates φ'_A as isogeny with kernel $(m_A)\varphi_B(P_A) + (n_A)\varphi_B(Q_A)$.
7. Alice gets $E_{AB} = \varphi'_A(E_B)$, which is isomorphic to Bob's $E_{BA} = \varphi'_B(E_A)$.

Algorithm (more technically)

Public paramters: Curve E_0 and points P_A, Q_A, P_B, Q_B.

Alice's real point of view

1. Alice chooses scalars m_A, n_A.
2. Alice calculates a point $R_A = (m_A)P_A + (n_A)Q_A$.
3. This uniquely determines isogeny φ_A such that $Ker(\varphi_A) = R_A$.
4. Alice sends E_A, $\varphi_A(P_B)$ and $\varphi_A(Q_B)$ to Bob.
5. Alice receives E_B, $\varphi_B(P_A)$ and $\varphi_B(Q_A)$.
6. Alice calculates φ'_A as isogeny with kernel $(m_A)\varphi_B(P_A) + (n_A)\varphi_B(Q_A)$.
7. Alice gets $E_{AB} = \varphi'_A(E_B)$, which is isomorphic to Bob's $E_{BA} = \varphi'_B(E_A)$.
8. They choose $j(E_{AB})$ as their shared secret.

Algorithm (idea)

1. Alice and Bob agree on a pool of curves and a starting curve E_0.
2. Alice applies a secret random sequence of **two-isogenies**, resulting in an elliptic curve $E_A = \varphi_A(E_0)$.
3. Bob secretly walks a random trail of **three-isogenies** in order to end up at a curve $E_B = \varphi_B(E_0)$.
4. They exchange curves E_A and E_B, and reapply the isogenies on the received curve.
5. They both end up with the same curve, E_{AB}.

Algorithm (idea)

two-isogenies

three-isogenies

Final notes

- Each party works with different isogenies.
- Set up during the choice of P_A, P_B, Q_A, Q_B.
- Choose primes l_A, l_B (most often 2 and 3).
- Choose p prime such that $p = (l_A^{e_A})(l_B^{e_B})f - 1$.
- We choose the field $\mathbb{F} = \mathbb{F}_{p^2}$.
- There are $(p+1)^2 = (l_A^{e_A} l_B^{e_B} f)^2$ points on the curve.
- Alice can use only isogenies s. t. $Ker(\varphi)$ is a $l_A^{e_A}$-torsion subgroup.

SIDH explained in a sci-fi manner `https://gist.github.com/defeo/163444a53252ba90cca6a3b550e6dd31` with a lot of spaceships

Thank you for your attention

Main source:

- David Urbanik's lecture "Introduction to the Post-Quantum Supersingular Isogeny Diffie-Hellman Protocol" at University of Waterloo: `https://www.youtube.com/watch?v=PW5Vsu57o9I`

Other sources:

- `https://blog.quantummadness.com/posts/supersingular-isogenies`
- `https://blog.cloudflare.com/sidh-go/`
- `https://crypto.anarres.info/2017/sidh`
- `https://www.esat.kuleuven.be/cosic/elliptic-curves-are-quantum-dead-long-live-elliptic-curves/`