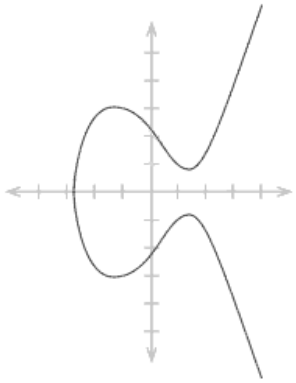

Figure 1: An elliptic curve, drawn over the reals.

1 Public key cryptography

Cryptography is a way for two people, referred to as Alice (A) and Bob (B), to communicate secretly over an insecure channel without an opponent, Oscar (O), understanding what is being said.

In a *public key* or *asymmetric* cryptosystem there are two keys. The *public key*, which is published in a directory and allows encryption, and the *private key* which is kept secret and allows decryption.

Algorithm 1.1 (ElGamal)**.** This algorithm allows two people to communicate messages secretly over an insecure communications channel.

1. (Setup) A finite cyclic group G of order n and generator $\alpha \in G$ are chosen. Each user picks a random integer $l \in \{0, 1, \ldots, n-1\}$ (the private key), and makes public α^l (the public key). We suppose that messages are elements of G and that user A wishes to send a message, m, to user B.
2. A generates a random integer $k \in \{0, 1, \ldots, n-1\}$ and computes α^k.
3. A looks up B's public key α^l and computes $(\alpha^l)^k$ then $m\alpha^{lk}$.
4. A sends to B the pair of group elements $(\alpha^k, m\alpha^{lk})$.
5. B computes $(m\alpha^{lk})((\alpha^k)^l)^{-1} = m\alpha^{lk}(\alpha^{lk})^{-1} = m$ and recovers the message.

Definition 1.2 (Discrete Logarithm problem)**.** Given a group G, an $\alpha \in G$ and α^l compute l.

Being able to solve the discrete logarithm problem allows one to crack the ElGamal cryptosystem. No efficient algorithm is known for computing discrete logarithms.

2 Elliptic curves

We can understand an elliptic curve as all the points on the curve $y^2 = x^3 + ax + b$ together with $\mathcal{O}$, the "point at infinity". We also require that the curve is non-singular, so has no cusps or self intersections. An example is shown in Figure 1,

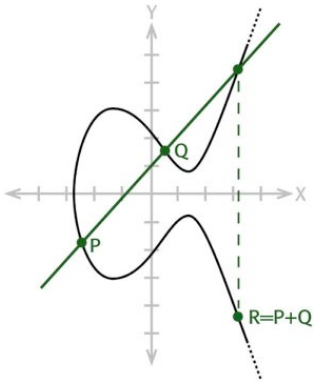

Figure 2: The group law on an elliptic curve.

where we consider the point at infinity to be the point infinitely far to the top and bottom of the graph.

To define the group law consider two points, P and Q, on our elliptic curve and draw the line from P to Q until you hit the curve again. This forms another point on the curve. Now draw a line from the point at infinity, $\mathcal{O}$, through this new point. The point where this line intersects the elliptic curve again is $P+Q$.

Algorithm 2.1. Let E be an elliptic curve given by $y^2 = x^3 + ax + b$.

(a) If $P_0 = \mathcal{O}$ then $-\mathcal{O} = \mathcal{O}$. Otherwise let $P_0 = (x_0, y_0) \in E$. Then $-P_0 = (x_0, -y_0)$.

(b) If one of P_1 or P_2 equals $\mathcal{O}$ then use $P + \mathcal{O} = \mathcal{O} + P = P$, otherwise let $P_1 + P_2 = P_3$ with $P_i = (x_i, y_i) \in E$.

If $x_1 = x_2$ and $y_1 = -y_2$ then $P_1 + P_2 = \mathcal{O}$.

Otherwise, let

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} \text{ and } \nu = \frac{y_1 x_2 - y_2 x_1}{x_2 - x_1} \text{ if } x_1 \neq x_2;$$

$$\lambda = \frac{3x_1^2 + a}{2y_1} \text{ and } \nu = \frac{-x_1^3 + ax_1 + 2b}{2y_1} \text{ if } x_1 = x_2.$$

(So $y = \lambda x + \nu$ is the line through P_1 and P_2, or tangent to E if P_1=P_2.)

Then $P_3 = (x_3, y_3)$ where $x_3 = \lambda^2 - x_1 - x_2$ and $y_3 = -\lambda x_3 - \nu$.

Theorem 2.2. *The addition law on an elliptic curve, E, has the following properties:*

1. *$P + \mathcal{O} = P$ for all $P \in E$.*
2. *$P + Q = Q + P$ for all $P, Q \in E$.*
3. *Let $P \in E$. There is a point of E, denoted $-P$, so that*
$$P + (-P) = \mathcal{O}.$$
4. *Let $P, Q, R \in E$. Then*
$$(P + Q) + R = P + (Q + R).$$

In other words, the addition law makes E into an abelian group with identity $\mathcal{O}$.

3 Elliptic curve cryptography

Using elliptic curves as the underlying group for public key cryptography was first suggested in 1985 by N. Koblitz and V. Miller because the discrete logarithm problem was believed to be harder for elliptic curves than for finite fields.

Algorithm 3.1 (ElGamal for Elliptic Curves)**.** We present our ElGamal algorithm again for elliptic curves. Note that is basically the same as before – the biggest difference is we are now writing our group additively.

1. (Setup) An elliptic curve, E, over a finite field, $\mathbb{F}_p$, is chosen, together with a point P on the curve. The point P generates a cyclic subgroup $G = \{\mathcal{O}, P, 2P, \ldots, (n-1)P\}$ of order n. Each user picks a random integer $l \in \{0, 1, \ldots, n-1\}$ (the private key), and makes public lP (the public key). We suppose that messages are elements of G and that user A wishes to send a message, m, to user B.

2. A generates a random integer $k \in \{0, 1, \ldots, n-1\}$ and computes kP.

3. A looks up B's public key lP and computes $k(lP)$ then $m + klP$.

4. A sends to B the pair of group elements $(kP, m + klP)$.

5. B computes $(m + klP) - l(kP) = m$ and recovers the message.

Once again being able to find l from P and lP (solving the *elliptic curve discrete logarithm problem*) would allow us to crack this cryptosystem.

Elliptic curve cryptography is more efficient than standard finite field cryptography: although the operations involved are more computationally intensive the smaller key size more than makes up for this. Elliptic curve cryptography is especially useful for low memory applications such as in smart cards.

4 Attacks on the discrete logarithm problem

1. Index calculus: Attack on discrete logarithm for $\mathbb{F}_p^\times$. Expected running time approximately $\exp\left(\sqrt{2 \log p \log \log p}\right)$.

2. Attacks on the discrete logarithm for general finite groups. These have expected running time approximately $\sqrt{p} = \exp\left(\frac{1}{2} \log p\right)$. They include Baby Step, Giant Step, Pollard's ρ method and Pollard's λ method.

3. Attacks on discrete logarithm for particular families of elliptic curves. Supersingular and anomalous curves are vunerable to attacks using the structure of these curves.