Elliptic curves in cryptography

INTRODUCTION

Since the 1980s, elliptic curves have been playing an increasingly important role both in number theory and in cryptography. They are used in some of the most widespread public key cryptosystems, they played an important role in the proof of Fermat's Last Theorem, and elliptic curve techniques have been developed for factorization and primality testing. The goal of this essay is to present the basic theory of elliptic curves and explain its application to cryptography.

THE BASIC THEORY

We define an elliptic curve E as the graph of an equation of the form

$$y^2 = x^3 + Ax + B,$$

where A and B are constants in some field K (then we say E is defined over K and denote the points of E as $E(K)$). We use this equation instead of a more general one involving other lower terms because if the field characteristic is neither 2 nor 3, then the general equation can always be reduced to this form, called the Weierstrass equation. It is not possible to draw meaningful pictures of elliptic curves over most fields, but it is useful to think in terms of graphs over the real numbers, which have two basic forms, depicted in Figure 1.

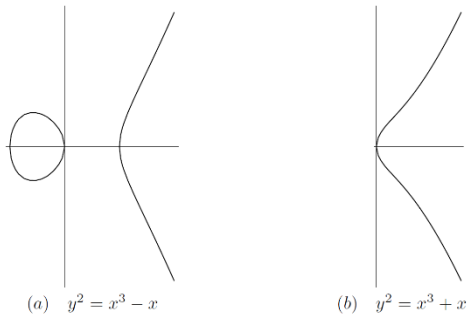

Figure 1: Two elliptic curves over the real numbers. **(Washington, 2003)**

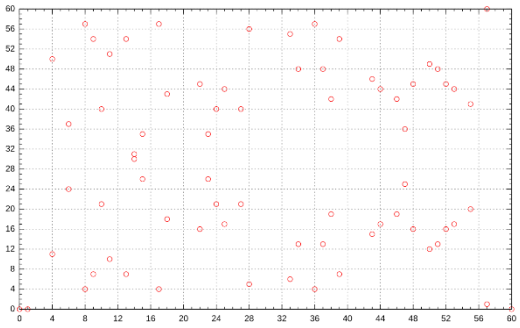

Figure 2: Elliptic curve $y^2 = x^3$-x on F_{61}. **(Wikipedia contributors, 2025)**

For technical reasons, a point at infinity is added to an elliptic curve. This concept can be made rigorous through projective geometry, but it can be intuitively regarded as a point at the top and bottom of the y-axis, as if the ends of the axis were wrapping around and meeting at this point. It can be thought of as a formal symbol satisfying certain rules (e.g. a line is said to pass through infinity when it is vertical).

One of the most important properties of and elliptic curve is that any line intersects it at most on three points. This allows the definition of an operation (addition) inside the set of points of an elliptic curve in the following way: if $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ are two points on E, define $P_1 + P_2$ to be the third intersection of the line L through P_1 and P_2 with the curve, reflected across the x-axis. We need to consider some special cases:

- If $x_1 = x_2$ but $y_1 \neq y_2$, the line through P_1 and P_2 is a vertical line, and so we define $P_1 + P_2 = \infty$.
- If $P_1 = P_2$, then we take L to be the line tangent to E on P_1. If $y_1 = 0$, then the line is vertical and the sum yields ∞ again, otherwise a new point can be obtained.
- If $P_2 = \infty$, then the line through P_1 and P_2 is a vertical line that intersects E in $-P_1$, so after reflecting we get that $P_1 + \infty = P_1$.

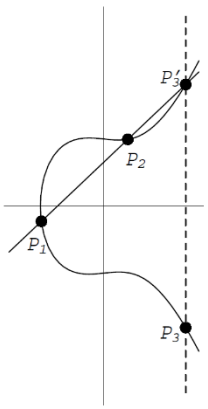

Figure 3: Adding points on an elliptic curve. **(Washington, 2003)**

This operation is commutative, associative, has an identity and each point has an inverse, so the points on E form an abelian group with ∞ as the identity element. This fact is called the group law of elliptic curves, and it is behind most of the interesting properties and applications of elliptic curves.

THE DISCRETE LOGARITHM PROBLEM

Let G be any group and let $a, b \in G$. Suppose that we know that $a^k = b$ for some integer k. The discrete logarithm problem is to find k. The RSA cryptosystem is based on the fact that for $G = \mathbb{Z}/N\mathbb{Z}$, with N a product of two primes, the discrete logarithm problem is equivalent to the problem of factoring an integer, which is very hard for large numbers. However, algorithms for factoring primes are getting increasingly better, so larger and larger primes are needed, increasing computational costs.

If we take $G = E(F_q)$, where F_q is a finite field, then the discrete logarithm problem is also very hard for appropriately chosen elliptic curves, which makes cryptosystems based on solving this problem very secure. In fact, elliptic curves provide security equivalent to classical systems while using fewer bits. For example, it is estimated that a key size of 4096 bits for RSA gives the same level of security as 313 bits in an elliptic curve system. This means that these systems can be implemented on devices with limited computational power.

ELLIPTIC CURVE CRYPTOGRAPHY

The basic setup for cryptography is that a person, Alice, wants to send a message to another person, Bob, without a third person, Eve, being able to read it. To do so, Alice uses an encryption key to encrypt the message and Bob uses a decryption key to decrypt it. There are two types of encryption. In symmetric encryption, the encryption and decryption keys are the same or one can be easily obtained from the other. In this case, Alice and Bob need a safe way to establish a key before sending the message. In public key encryption, however, Alice and Bob do not need to have prior contact: Bob publishes an encryption key, which Alice uses, while keeping secrete the decryption key. This is secure if the decryption key cannot be deduced from the encryption key. This is where the discrete logarithm problem comes in.

Generally, public key systems are slower than good symmetric systems. Therefore, it is common to use a public key system to establish a key that is then used in a symmetric system. This can be done in the following way:

1. Alice and Bob agree on an elliptic curve E over a finite field F_q such that the discrete logarithm problem is hard in $E(F_q)$. They also agree on a point $P \in E(F_q)$ with a large order.
2. Alice chooses a secret integer a, computes aP, and sends it to Bob.
3. Bob chooses a secret integer b, computes bP, and sends it to Alice.
4. They both use their secret integers to compute $abP = a(bP) = b(aP)$.
5. Alice and Bob use some publicly agreed on method to extract a key from abP. For example, they could use as key the last 256 bits of the x-coordinate of abP.

The only information that Eve sees are E, F_q, P, aP and bP. If Eve can solve discrete logs in $E(F_q)$, then she can compute abP, and it is not known whether there is another way to compute it.

The are also methods to encrypt messages using elliptic curves (Massey-Omura encryption, ElGamal encryption, ECIES) and to generate digital signatures (ElGamal digital signatures and ECDSA).

REFERENCES

Washington, L. C. (2003). *Elliptic Curves: Number Theory and Cryptography. Second Edition.* CRC Press.

Wikipedia contributors. (2025, January 1). *Elliptic curve*. In *Wikipedia, the Free Encyclopedia*. https://en.wikipedia.org/wiki/Elliptic_curve