

Contents

1	Mathematical Background	1
1.1	Abstract Algebra	1
	Groups	1
	Abelian Groups	2
	Cyclic Groups	2
	Rings	2
	Commutative Rings	2
	Integral Domains	2
	Fields	3
	Field Extensions and Splitting Fields	3
	Field Automorphisms	3
	Modular Arithmetic	4
	Galois Fields in Cryptography	5
2	Public Key Cryptography	7
2.1	Elliptic Curve Cryptography (ECC)	8
	Purpose	8
	Basic Maths	8
	Prime Curves	11
	Binary Curves	12
	Cryptography with Elliptic Curves	12
	ECC Security	13

1 Mathematical Background

1.1 Abstract Algebra

This section introduces fundamentals of number theory or ‘abstract algebra’. It is necessary to understand the basic concepts of groups, rings, and fields because cryptography heavily depends on them. For your interest, some supporting ideas that lead to Galois fields are shortly reviewed. The concept of Galois fields allows deep analysis of the structure and the solutions of polynomials, however (fortunately) for our purposes only Galois fields of the form $\text{GF}(2^m)$ are important.

Groups

A *group* G is a set of elements with a binary operation such as ‘+’ (or ‘.’ or whatever operation we might define) for which the following axioms are true:

1. If the elements x and y belong to G then also the element $x + y$ belongs to G (‘closure’).
2. There is an *identity* element e in G such that $x + e = x = e + x$.
3. The concatenation of operations does not lead to ambiguous results. That is $x + (y + z) = (x + y) + z$ for all x, y, z in G . The order of the operations does not matter (‘associative’).
4. Every element x in G has a *unique inverse element* x' so that $x + x' = e = x' + x$. (The inverse element can also be written as x^{-1} or $-x$.)

For example the set of all integers (positive, negative, and zero) form a group ‘under addition’. The notation for this group is $(\mathbb{Z}, +)$. Obviously this is an *infinite group* because the number of elements (the ‘order’) is infinite. The identity element is 0 because $x + 0 = x$, and the negative integers are the inverse elements of the positive (and vice versa) because $x + (-x) = 0$. Obviously the addition in \mathbb{Z} is also associative.

infinite group

Note that \mathbb{Z} under multiplication does *not* form a group because the inverse elements to get the identity 1 are fractional numbers.

Another important group example is the *symmetric group* S_n of all permutations of n elements $\{1, 2, 3, \dots, n\}$. The permutations π and ρ are elements of S_n and at the same time group operations. That is the composite mapping $\pi \cdot \rho$ means ‘permute the elements of ρ according to the permutation π ’.¹ For example if $\pi = \{3, 2, 1\}$ and $\rho = \{2, 1, 3\}$

symmetric group

¹In this case π is applied on ρ and therefore π can be regarded as the operator.

then $\pi \cdot \rho = \{3, 1, 2\}$ which is obviously also an element of S_n . The identity element $\eta = \{1, 2, 3\}$ as operator maps any element to itself, so $\eta \cdot \rho = \rho$. Obviously, there is always an inverse element π^{-1} which reverses the permutation π .

Abelian Groups

An *abelian group* is a group for which the group operation is *commutative*. The previous group example $(Z, +)$ is actually an abelian group.

Cyclic Groups

A *cyclic group* can be *generated* by a single element, called the *generator* of the group. Every cyclic group is also an abelian group. A finite cyclic group is also called a *periodic group*.

For example the n roots of unity $e^{i2\pi k/n}$ with $k = \{0, \dots, n-1\}$ form a periodic cyclic group under multiplication with $\mu = e^{i2\pi/n}$ as generator and $\mu^0 = 1$ as the identity element ($k = 0$). The elements of the cyclic group are $\{\mu^0, \mu^1, \mu^2, \dots, \mu^{n-1}\}$.

Rings

A *ring* R is an abelian group for one binary operation (e. g. the addition '+') and allows a second binary operation, for example the multiplication '×'. The following properties must be valid for this second operation:

1. $x \times y$ is also an element of R (closure).
2. $(x \times y) \times z = x \times (y \times z)$, hence the operation is associative.
3. $x \times (y + z) = x \times y + x \times z$, hence both operations are distributive.

Typically the notation of a ring is $(R, +, \times)$.

Commutative Rings

As the name implies, a *commutative ring* is a ring for which both operations are commutative.

Integral Domains

An *integral domain* is a commutative ring with

1. A multiplicative identity
2. No zero divisors: if $a \neq 0$ and $b \neq 0$ then $ab \neq 0$.²

²A so-called *zero divisor* $b \neq 0$ would lead to $ab = 0$. A good example is vector multiplication: let $a = (0, 1)$ and $b = (1, 0)$ then $a \cdot b = (0, 0)$, so both a and b are zero divisors.

Fields

A *field* is an integral domain with a multiplicative inverse. In summary, a field is simply a set with two operations, with identity and inverse elements, and the laws of associativity, commutivity, and distribution. In even simpler words, a field is like a group with an additional operation.

Simple examples of fields are the rational, real, and complex numbers. The set of integer numbers is not a field because there is no multiplicative inverse.

Field Extensions and Splitting Fields

Another important type of fields are typically notated like $Q[\sqrt{2}]$ which (in this case) contains all numbers that can be written as $a + b\sqrt{2}$ with a, b as rational numbers. Obviously the inverse element $1/(a + b\sqrt{2})$ exists and can also be written in this form: $c + d\sqrt{2}$.

Now consider polynomials. We call α an *algebraic number* if $p(\alpha) = 0$ for some appropriate polynomial $p(x)$. In other words, α is a zero-point or root of $p(x)$. If α is an algebraic number then $Q[\alpha]$ is a field. The field $Q[\alpha]$ can be regarded as the set of elements $a_0 + a_1\alpha + \dots + a_{n-1}\alpha^{n-1}$ (again a_i are rational numbers) if n is the smallest degree of a polynomial $p(x)$ for which $p(\alpha) = 0$. For example, since $\alpha = \sqrt[4]{2}$ is one zero point of the polynomial $x^4 - 2 = 0$ of order $n = 4$, then $Q[\alpha]$ is a field with the elements $\{a_0 + a_1\sqrt[4]{2} + a_2\sqrt[4]{2}^2 + a_3\sqrt[4]{2}^3\}$ with $a_i \in \mathbb{R}$.

algebraic number

Also the inclusion of multiple algebraic numbers is possible. For example, since $\alpha_{1,2} = \pm\sqrt{2}$ and $\alpha_{3,4} = \pm\sqrt{3}$ are the zero points of the polynomial $p(x) = x^4 - 5x^2 + 6$ of order $n = 4$, then $Q[\alpha_1, \alpha_3]$ is a field with the elements³ $\{a_0 + a_1\sqrt{2} + a_2\sqrt{3} + a_4\sqrt{2}\sqrt{3}\}$.

The fields $Q[\sqrt{2}]$, $Q[\sqrt[4]{2}]$, etc. are so-called *field extension* of the field Q , typically notated as $Q \supseteq Q[\sqrt{2}]$. That is, fields like $Q[\sqrt{2}]$ are supersets of Q because every element x of Q can be written as element of $Q[\sqrt{2}]$ such as $x + 0\sqrt{2}$.

Moreover if a field extension of Q contains *all* zero points of $p(x)$ this field is also called a *splitting field*. For example $Q[\sqrt{2}, \sqrt{3}]$ is the splitting field of $p(x) = x^4 - 5x^2 + 6$.

Field Automorphisms

A *field automorphism* is a function f that maps the elements of a field to other elements of this field in a unique way such as $f(a_0 + a_1\sqrt{2}) = a_0 - a_1\sqrt{2}$. Furthermore

$$\begin{aligned} f(x + y) &= f(x) + f(y) \\ f(ax) &= f(a)f(x) \\ f(1/x) &= 1/f(x) \end{aligned}$$

are the required properties of valid automorphisms. The best way to think of automorphisms is that they actually *re-label* the elements of a field without changing the structure. For example taking the inverse is a valid automorphism in $R \setminus \{0\}$.

³Actually all algebraic combinations are possible. If only a single algebraic number is given, then a simple polynomial development is the only possibility.

Since automorphisms are invertible $f(p(x)) = p(f(x))$ (if f is a field automorphism of a splitting field of $p(x)$). Therefore if α is a root of $p(x)$ then also $f(\alpha)$ is a root because $p(f(\alpha)) = f(p(\alpha)) = f(0) = 0$.

For example if $p(x) = x^2 - 2$ then the splitting field is $Q[\sqrt{2}]$ and the Q -automorphisms are (always) the group identity $f_0(x) = x$ and $f_1(a_0 + a_1\sqrt{2}) = a_0 - a_1\sqrt{2}$.

Galois group

Now a Galois group G (or 'Gal(p)') is simply a collection of Q -automorphisms of a splitting field. Furthermore, if $f_0, f_1 \in G$ then also $f_0 \cdot f_1 \in G$ and $(f_0 \cdot f_1)(x) = f_0(f_1(x))$. Interestingly the square of one automorphism is the identity automorphism, $f_1 \cdot f_1 = f_0$ because

$$\begin{aligned} (f_1 \cdot f_1)(a_0 + a_1\sqrt{2}) &= f_1(f_1(a_0 + a_1\sqrt{2})) \\ &= f_1(a_0 - a_1\sqrt{2}) \\ &= a_0 + a_1\sqrt{2} \\ &= f_0(a_0 + a_1\sqrt{2}) \end{aligned}$$

and these is a property of a cyclic symmetric group (in this case of order 2).

Exercise: let f_1 be any permutation of the symmetric group S_n and apply $f_1 \cdot f_1$ on another permutation; show that $f_1 \cdot f_1$ is equal to the identity element $\{1, 2, 3, \dots\}$.

Modular Arithmetic

As described in the sections above, the result of arithmetic operations with elements of a finite field is also an element of that field. In this chapter mainly fields of prime order with integers as elements are investigated, that is $F = \{0, 1, 2, \dots, p - 1\}$ and p is a prime.

modulo operation

Practically, if numbers become larger than the order of the field minus one, they must 'wrap around'—similarly as when counting the hours on a clock face.⁴ The resulting element can easily be found using the *modulo* operation $a \bmod p = r$, for example $34 \bmod 7 = 6$. The *residue* r allows to write any integer a as $a = p \cdot \lfloor a/p \rfloor + (a \bmod p)$.

congruence

Clearly, for every integer in the field there are infinite integers outside the field which can be 'mapped back' to it. All these integers are said to be *congruent modulo* p . For example the integers 6, 13, 20, 27, 34, ... are all congruent modulo 7 because

$$\begin{aligned} 13 \bmod 7 &= 6 \\ 20 \bmod 7 &= 6 \\ 27 \bmod 7 &= 6 \\ 34 \bmod 7 &= 6 \\ &\dots \end{aligned}$$

and therefore congruency is often written as identity relation, e. g. $13 \equiv 34 \bmod 7$. It is also said that the elements form a *set of residues* and that each element represents a *residue class* modulo p . For example the numbers $3 + 7k$ with $k \in \mathbb{Z}$ that is

residue classes

⁴Euler initially called this 'Uhrenarithmetik'

$\{\dots, -18, -11, -4, 3, 10, 17, 24, \dots\}$ belong to the residue class $3 \pmod{7}$.

The *additive inverse* of the element 6 is 1 because $6 + 1 = 0 \pmod{7}$. Obviously every element has an additive inverse (hence subtraction is possible).

If we want every element to have a *multiplicative inverse* then p must be prime. Generally, if an element a and the order p are not *relative prime* (i. e. don't have common factors other than 1) then a cannot 'generate' all the other elements by multiplication and it also has no multiplicative inverse. In other words, $a \cdot k \pmod{p}$ with $k \in \mathbb{Z}$ will not produce all residues of the set.

relative prime

The following examples should illustrate this:

- If $p = 7$ the multiplicative inverse of the element 6 is also 6 because $6 \cdot 6 \equiv 1 \pmod{7}$. The multiplicative inverse of 3 is 5 because $3 \cdot 5 \equiv 1 \pmod{7}$.
- But if p is *not* prime, for example $p = 6$, then only the elements 1 and 5, which are relative prime to 6, have a multiplicative inverse (again 1 and 5 respectively), so the elements do not form a field.
- The numbers 2,3,4 are not relative prime to $p = 6$ and therefore they have no unique multiplicative inverse. But without an unique multiplicative inverse, the equation

$$4 \cdot x \equiv 4 \cdot y \pmod{6}$$

does not result in $x \equiv y \pmod{6}$ (in this case $x = 1$ and $y = 4$). In other words, multiplication is not injective (a one-to-one mapping).

Obviously, if p is prime then every element of the set is relative prime to p .

Galois Fields in Cryptography

Finite fields of order p^n , with p prime and n any positive integer, are traditionally called *Galois Fields*, $\text{GF}(p^n)$. In the previous section we already showed that finite sets of order p have field properties, essentially because an unique multiplicative inverse exists. So how can a set of order p^n be a field? We'll see that the elements are polynomials rather than numbers.

In cryptography all operations should be applied on integer sets whose order is a power of two. Basically we are interested in arithmetic operations on bytes. But the number set defined by a byte is $\{0, \dots, 255\}$ and the order is 256. Hence division is not possible. Even worse multiplication with these elements does not lead to unique results. Element which are relative prime to 256 produce ambiguous products, for example

$$16 \cdot 43 \equiv 16 \cdot 27 \pmod{256}.$$

Clearly it would be inefficient to calculate with sets of prime order (e. g. $\text{F}(251)$ or $\text{F}(257)$) because some bit combinations would not be used.

On the other hand, it can be shown that *polynomial arithmetic* including division is possible if the coefficient set is a field.⁵ Generally, if the order of the polynomials is

⁵The set of polynomials form a *polynomial ring*.

$n - 1$ and the coefficient set is of order p , then the set of polynomials is of order p^n and have the general form

$$P(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \dots + a_1x + a_0 = \sum_{i=0}^{n-1} a_i x^i .$$

However, polynomial division is often not exact, and then we get a quotient and an *irreducible* remainder polynomial:

$$\begin{aligned} \frac{P(x)}{D(x)} &= Q(x) + \frac{R(x)}{D(x)} \quad , \text{ or} \\ P(x) &= Q(x)D(x) + R(x) . \end{aligned}$$

prime polynomial An irreducible polynomial is also called a *prime polynomial*.

Interestingly, if p is prime and n arbitrary, the set of polynomials is a finite field if

- Addition and subtraction on the coefficients is performed modulo p
- Multiplication is performed modulo a prime polynomial if the product polynomial is of greater degree than $n - 1$

Generally it is said that ‘arithmetic is performed under $\text{GF}(p^n)$ ’.

GF(2) For digital cryptography, polynomial arithmetic over $\text{GF}(2)$ is especially practical because addition and subtraction are equivalent to the logical XOR operation and multiplication is equivalent to the logical AND operation—both operations are efficiently implemented in all CPUs.

AES For example, the AES cipher uses arithmetic in the finite field $\text{GF}(2^8)$ with the prime polynomial $P_{AES}(x) = x^8 + x^4 + x^3 + x + 1$. Arithmetic operations of two polynomials such as $P_1(x) = x^5 + x^3 + x + 1$ and $P_2(x) = x^4 + x^3 + 1$ lead to the following results:

$$\begin{aligned} P_1(x) \pm P_2(x) &= x^5 + x^4 + x + 1 \\ P_1(x) \cdot P_2(x) &= x^9 + x^8 + x^7 + x^6 + x + 1 \\ P_1(x) \cdot P_2(x) \text{ mod } P_{AES} &= x^7 + x^6 + x^5 + x^3 + x^2 + x . \end{aligned}$$

Therefore the result of the multiplication is an irreducible polynomial and its digital representation is 11101110, that is every bit represents one coefficient.

As with numbers, every polynomial P_i in $\text{GF}(p^n)$ can be considered as a residue class. By the way, as mentioned in a previous section (Field Automorphisms), Galois Fields of the same order always have the same structure even when the elements are differently labeled (numbers, polynomials, permutations, etc).

2 Public Key Cryptography

2.1 Elliptic Curve Cryptography (ECC)

Purpose

Recently¹ 768-bit RSA keys had been successfully attacked and therefore also 1024 bit keys are no longer considered to be secure. Actual security requirements demand RSA key lengths of at least 2048 bits. The involved computation efforts causes high CPU loads and long processing delays which is an undesirable challenge for low-power devices such as PDAs or mobile phones.

smaller key lengths

ECC provides a relatively new public-key method which is supposed to provide similar security levels at much smaller key lengths. Despite this the RSA algorithm is still the dominating method today because ECC is simply too 'young'. Recently ECC had been standardized as IEEE P1363.

Basic Maths

cubic equation

Actually an elliptic curve is not really an ellipse. Rather the *cubic* equation

$$y^2 + axy + by = x^3 + cx^2 + dx + e \quad (2.1)$$

looks only similar as those equations which are used to calculate the circumference of ellipses. Actually, an elliptic curve is a function of two variables and some coefficients, all of which defined over a finite field. Usually the elliptic curve is regarded as a *set* $E(a, b, c, d, e)$ which is defined only by its coefficients and consists of all points (x, y) that satisfy equation (2.1). For example the simplified elliptic curve

$$y^2 = x^3 + ax + b = E(a, b) \quad (2.2)$$

Abelian group

is an Abelian group (that is a group where the addition is commutative), if it has unique factors. Using Cardano's discriminant this leads to the requirement

$$4a^3 + 27b^2 \neq 0.$$

Figure 2.1 shows examples of elliptic curves of type $E(a, b)$.

Group Definitions

The following definitions support the addition of points on the elliptic curve:

- **Negative point:** The negative of a point A is the point with the same x coordinate but the negative of the y coordinate. So if $A = (x, y)$ then $-A = (x, -y)$. Note that both points lie on a vertical line.
- **Additive identity:** As group requirement, there is an additive identity O which means $A + O = A$. The point O is also called *zero point* or (confusingly) *point at infinity*.
- **Addition:** When some points A and B on the elliptic curve are interconnected by a line, there is a third intersection C .

¹See [Datenschleuder 81, Chaos Computer Club](#)

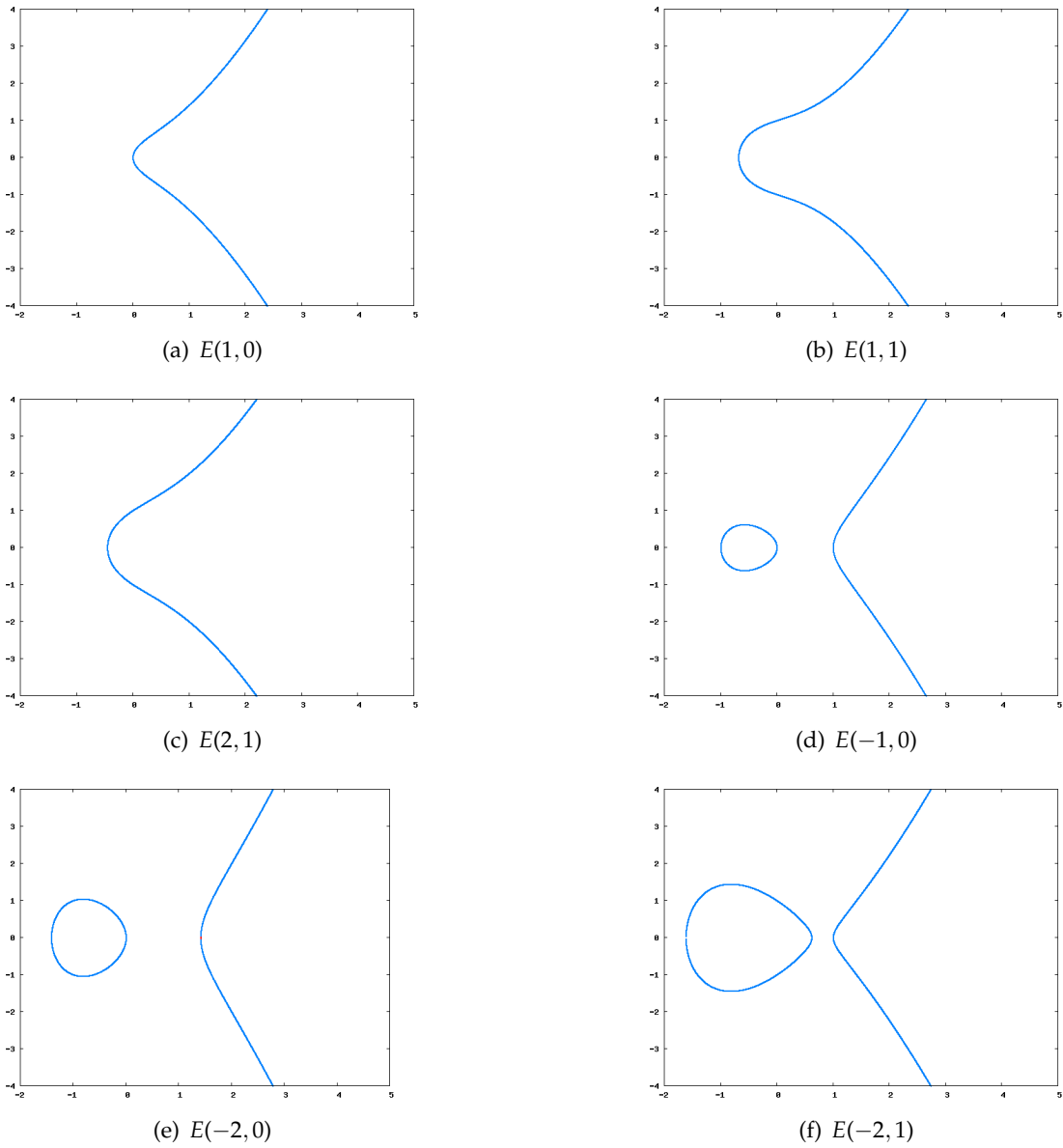


Figure 2.1: Elliptic curves $E(a, b) = y^2 = x^3 + ax + b$ in \mathbb{R} with different parameters a, b

- The addition is *defined* as $A + B = -C$, that is the addition $A + B$ is the negative of the third point C .
- Obviously, if three points lie on a straight line their sum is O .
- On a vertical line the third point is the point at infinity: $A + (-A) = 0$.
- In order to double a point A a tangent must be drawn at that point and the second intersection is the negative of $2A$.

Figure 2.2 shows the addition graphically.

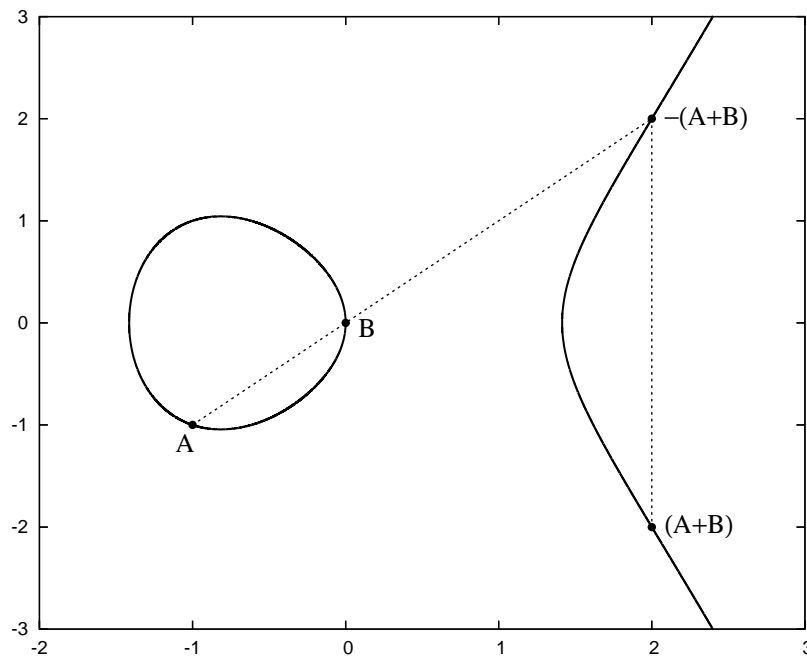


Figure 2.2: Addition in $E(-2, 0)$

For numerical computations an algebraic description is necessary. Using the slope $\Delta = \frac{y_B - y_A}{x_B - x_A}$ between A and B , the point C can be calculated according

$$\begin{aligned} x_C &= \Delta^2 - x_A - x_B \\ y_C &= -y_A + \Delta(x_A - x_C) . \end{aligned}$$

In case of doubling a point, e. g. $C = 2A$ the equations are

$$x_C = \left(\frac{3x_A^2 + a}{2y_A} \right)^2 - 2x_A$$

$$y_C = \left(\frac{3x_A^2 + a}{2y_A} \right)(x_A - x_C) - y_A$$

In this section, elliptic curve arithmetic has been introduced on real numbers. However, practical Elliptic Curve Cryptography operates on finite fields with positive numbers. There are two important families of elliptic curves:

1. **Prime curves** are defined over a finite field Z_p and can be efficiently implemented in software. Basically all numbers (x, y, a, b) are taken modulo a prime p .
2. **Binary curves** are defined over $GF(2^n)$ and can be efficiently implemented in hardware.

Prime Curves

In this section we investigate prime curves of the form

$$y^2 \pmod p = (x^3 + ax + b) \pmod p .$$

Such 'curves' consist of a set of points (x, y) which are basically symmetric about $y = p/2$. Again this elliptic curves only form an Abelian group if the discriminant is non-zero

$$(4a^3 + 27b^2) \pmod p \neq 0 .$$

Also the arithmetic rules are similar as with the real numbers:

- $A + 0 = A$
- If $A = (x_A, y_A)$ then $-A = (x_A, -y_A)$
- $A + (-A) = 0$
- The point $C = A + B$ can be calculated using $\delta = \frac{y_B - y_A}{x_B - x_A} \pmod p$ according

$$x_C = (\delta^2 - x_A - x_B) \pmod p$$

$$y_C = [\delta(x_A - x_B) - y_A] \pmod p .$$

The above formulas can also be used if a single point should be doubled, that is $A = B$, but in this case $\delta = \frac{3x_A^2 + a}{2y_A} \pmod p$.

- Multiplication is repeated addition

If p is large, the number of points n on the elliptic curve is approximately equal to the number of elements in the field Z_p . Actually, n is bounded by

$$1 - 2\sqrt{p} + p \leq n \leq 1 + 2\sqrt{p} + p$$

Binary Curves

For binary curves all numbers (x, y, a, b) are elements of $\text{GF}(2^n)$ and the elliptic curve equation must be of the form

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

This elliptical curve is only an Abelian group if $b \neq 0$. Again the arithmetic rules are similar as in the previous cases plus some exceptions:

- $A + 0 = A$
- If $A = (x_A, y_A)$ then $-A = (x_A, x_A + y_A)$
- $A + (-A) = 0$
- The point $C = A + B$ can be calculated using $\delta = \frac{y_B - y_A}{x_B - x_A}$ according

$$\begin{aligned} x_C &= \delta^2 + \delta + x_A + x_B + a \\ y_C &= \delta(x_A + x_C) + x_C + y_A. \end{aligned}$$

- If a point should be doubled, that is $A = B$, then $\delta = x_A + y_A/x_A$ and

$$\begin{aligned} x_C &= \delta^2 + \delta + a \\ y_C &= x_A^2 + (\delta + 1)x_C. \end{aligned}$$

Cryptography with Elliptic Curves

In order to utilize elliptic curve arithmetics for cryptography one has to find a 'hard problem', i. e. an operation for which the inverse operation is technically unfeasible. Such hard problem is the *discrete logarithm problem for elliptic curves*:

discrete logarithm

Given A and $B = \lambda A$, find λ .

In other words, 'find the discrete logarithm λ of B to the base A ', when B and A are points on an elliptic curve. The brute force method would be to try all multiples of A until B results, but this is infeasible for large fields and large λ .

DH equivalent

A Diffie-Hellman algorithm using ECC works as follows:

1. Choose a *base point* G of large *order* n . The 'order of a point' on an elliptic curve is the smallest positive integer n such that $nG = 0$. The elliptic curve $E_p(a, b)$ and the base point G are publicly known.
2. User A chooses a private key² $S_A < n$ and calculates the public key $P_A = S_A \cdot G$
3. User B chooses a private key $S_B < n$ and calculates the public key $P_B = S_B \cdot G$

²S for secret

4. Both A and B exchange their public keys
5. User A calculates the common secret key $C = S_A \cdot P_B = S_A \cdot S_B \cdot G$
6. User B calculates the common secret key $C = S_B \cdot P_A = S_B \cdot S_A \cdot G$

Note that every number above is actually a point and consists of an x - and y -component. A practical algorithm might choose only one component as key.

On the other hand, an RSA algorithm using ECC works as follows:

RSA equivalent

1. Goal: 'encrypt' the point M on the elliptic curve. Of course there must be a well defined mapping between real messages and points on the curve. Again a common base point G is needed by both parties.
2. User A chooses a private key S_A and calculates a public key $P_A = S_A \cdot G$
3. User B chooses a private key S_B and calculates a public key $P_B = S_B \cdot G$
4. User A chooses a random λ for the particular encryption and calculates the ciphertext $C_M = (X, Y) = (\lambda G, M + \lambda P_B)$ which is actually a pair of points. Note that the second point $Y = M + \lambda P_B$ contains the message M but it is 'masked' by λP_B . The first point $X = \lambda G$ is carried as a 'clue' how to decrypt the ciphertext.
5. User B now takes the Y -component of C_M and subtracts $S_B \cdot X$, that is:

$$\begin{aligned}
 Y - S_B \cdot X &= M + \lambda P_B - S_B(\lambda G) \\
 &= M + \lambda(S_B \cdot G) - S_B(\lambda G) \\
 &= M
 \end{aligned}$$

ECC Security

Even the fastest known EC-logarithm problem solver, the *Pollard rho method*, is remarkably inefficient.

Example: The estimated efforts to break RSA with the currently recommended key size of 2048 bits, using the general number field sieve is approximately 3×10^{20} years. But breaking ECC with a key size of only 234 bits, using the Pollard rho method, requires 1.6×10^{28} years!