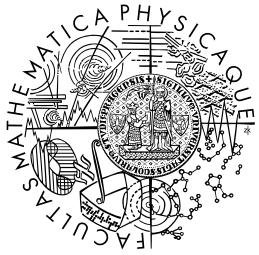

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

**BACHELOR THESIS**

Igor Eržiak

# Counting the points on elliptic curves over finite fields

Department of Algebra

| | |
|---|---|
| Supervisor of the bachelor thesis: | doc. RNDr. Jan Šťovíček, Ph.D. |
| Study programme: | Mathematics |
| Study branch: | Mathematical Methods of Information Security |

Prague 2018

I declare that I carried out this bachelor thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In ..... date .....  signature of the author

I would like to thank my supervisor doc. RNDr. Jan Šťovíček, Ph.D for his patience and help throughout the process of creating this thesis. In addition, I want to thank my friend Martin Žurav who has provided me with a great support and excellent advice while I was struggling with this thesis.

Title: Counting the points on elliptic curves over finite fields

Author: Igor Eržiak

Department: Department of Algebra

Supervisor: doc. RNDr. Jan Šťovíček, Ph.D., Department of Algebra

Abstract: The goal of this thesis is to explain and implement Schoof's algorithm for counting points on elliptic curves over finite fields. We start by defining elliptic curve as a set of points satisfying certain equation and then proceeding to define an operation on this set. Theoretical background needed for the algorithm is presented in the second chapter. Finally, the Schoof's algorithm is introduced in the third chapter, supplemented by an implementation in SageMath open-source software.

Keywords: Schoof's algorithm, Elliptic curve, Division polynomial, Frobenius endomorphism

# Contents

# Introduction

Elliptic curves play an important role in modern algebra. Not only have they been used in proof of Fermat's Last Theorem by Andrew Wiles but they provide an important tool for public-key cryptography, too. Moreover, they are used in primality testing and factorization algorithms.

In cryptography, it is always important to know as much as possible about the structure of the group we are working with. One of the key properties is the order of the group. Up until 1985, there was no deterministic polynomial-time algorithm known to compute the order of the elliptic curve group. However, this had changed when René Schoof introduced his algorithm, that used Frobenius map properties and division polynomials, combining the information gained using the well-known Chinese Remainder Theorem.

This thesis will aim to explain Schoof's approach and provide the theory required to understand it. An important part will be an implementation in high-level open-source mathematical software SageMath.

# 1. Elliptic curves

The goal of this chapter is to introduce the notion of an elliptic curve. We will start with a set of solutions of a certain equation and define an operation on it.

## 1.1 Definition

For purposes of this thesis, we define the elliptic curve as follows:

**Definition 1.1.** *Let $\mathbb{F}$ be a field with $char(\mathbb{F}) \neq 2, 3$ and let $A, B \in \mathbb{F}$ satisfying $4A^3 + 27B^2 \neq 0$. **Elliptic curve** $E(\mathbb{F})$ defined by the equation*

$$y^2 = x^3 + Ax + B \tag{1.1}$$

*is the set*

$$E(\mathbb{F}) = \{\mathcal{O}\} \cup \left\{(x, y) \in \mathbb{F} \times \mathbb{F} \mid y^2 = x^3 + Ax + B\right\},$$

*where $\mathcal{O}$ is called the point at infinity.*

*Note.* Equation of the form (1.1) is called the **short Weierstrass equation** of an elliptic curve.

There is a couple of points in the definition that will be addressed briefly. For more detailed explanation see Washington [2008], Section 2.1.

In general, elliptic curves can be defined over any field $\mathbb{F}$ using the generalized Weierstrass equation

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_5$$

with $a_1, \ldots, a_5 \in \mathbb{F}$. However, if $char(\mathbb{F}) \neq 2, 3$, all elliptic curves can be defined by a short Weierstrass equation (see Silverman [2009], Chapter III §1.). This simplifies all computations without loss of generality. Moreover, there are more efficient algorithms than Schoof's algorithm in case when $char(\mathbb{F})$ is small (see Washington [2008], Section 4.3.1).

Furthermore, elliptic curves can be equivalently defined as non-singular plane algebraic curves given by equation (1.1). Non-singularity means that the polynomial $x^3+Ax+B$ has no multiple roots. This happens if and only if $4A^3+27B^2 \neq 0$. Consequently, it allows the operation on the elliptic curve to be well-defined.

Finally, let us look into the projective space consisting of all triples $(x, y, z)$ with $x, y, z \in \mathbb{F}$ and at least one of them nonzero. We have an equivalence relation $(x_1, y_1, z_1) \sim (x_2, y_2, z_2)$ if there is an element $\lambda \in \mathbb{F}$ such that

$$(x_1, y_1, z_1) \sim (\lambda x_2, \lambda y_2, \lambda z_2).$$

If we make the equation (1.1) homogeneous by introducing the third variable $z$, we have

$$y^2z = x^3 + Axz^2 + Bz^3.$$

The points with $z = 1$ correspond to the set

$$\left\{(x, y) \in \mathbb{F} \times \mathbb{F} \mid y^2 = x^3 + Ax + B\right\}$$

while setting $z = 0$ leads to

$$x^3 = 0$$

with the only nonzero solution, up to the equivalence $\sim$, being $(0, 1, 0)$. The points with $z = 0$ are usually interpreted as points at infinity. Since elliptic curves have only one such point, we will refer to it as *the* point at infinity and denote it by $\mathcal{O}$. For more details see Washington [2008], Section 2.3.

## 1.2 Group operation

Let us consider the elliptic curve $E(\mathbb{R})$ defined by the equation

$$y^2 = x^3 - 2x + 4. \tag{1.2}$$

Plugging in a couple of values for $x$ and $y$ we can easily find some points on the curve $E(\mathbb{R})$, for example:

$$(-2, 0), (0, \pm 2), (1, \pm\sqrt{3}).$$

Note that for every solution $(x, y)$ the point $(x, -y)$ is also a solution.

We can use two known points on the curve to produce a new point. Start with the points $(-2, 0)$ and $(0, 2)$. Consider a straight line through these points, defined by the equation

$$y = x + 2. \tag{1.3}$$

Substituting this for $y$ in the equation (1.2) and rearranging the terms we have

$$x^3 - x^2 - 6x = 0.$$

Knowing two roots, $x_1 = -2$ and $x_2 = 0$, we could factor the polynomial to find the third. However, there is a simpler way. If $x_1, x_2, x_3 \in \mathbb{R}$ are the roots of a third degree monic polynomial then

$$(x - x_1)(x - x_2)(x - x_3) = x^3 - (x_1 + x_2 + x_3)x^2 + \text{lower degree terms}.$$

Comparing the coefficients at $x^2$ we have

$$1 = x_1 + x_2 + x_3 = -2 + 0 + x_3$$

$$\implies x_3 = 3.$$

Plugging the value of $x_3$ into (1.3) we have

$$y_3 = 3 + 2 = 5.$$

We have produced a new point on the curve, the point $(3, 5)$. In fact, from the symmetry in $y$, we have produced one more point, namely $(3, -5)$.

The idea of starting with two points on the curve and using the line passing through them to find a new point gives rise to the idea of an operation on $E(\mathbb{F})$. However, if we defined the sum of two points $P, Q \in E(\mathbb{F})$ on the line intersecting

the curve to be the third point of intersection, say $R$, we would not get very far. We would have

$$P + Q = R \text{ and } P + R = Q$$
$$\implies P + P + Q = Q.$$

This would mean that for any $P \in E(\mathbb{F})$ the order of $P$ would be at most 2.

To avoid this, we always reflect the resulting point over the $x$-axis. In the example above we would have

$$(-2, 0) + (0, 2) = (3, -5).$$

In the situation when we want to add the point to itself, the idea is very similar. However, since we only have one point to define a line we take the tangent line to find the new point.

The following definition describes this operation formally.

**Definition 1.2** (Group operation)**.** *Let $E(\mathbb{F})$ be an elliptic curve defined by $y^2 = x^3 + Ax + B$. Let $P, Q \in E(\mathbb{F})$, $P, Q \neq \mathcal{O}$, where $P = (x_1, y_1)$ and $Q = (x_2, y_2)$. We define the sum $P + Q$ as follows:*

- *If $x_1 \neq x_2$, then $P + Q = R = (x_3, y_3)$ where*

$$x_3 = m^2 - x_1 - x_2, \quad y_3 = m(x_1 - x_3) - y_1, \quad m = \frac{y_2 - y_1}{x_2 - x_1}.$$

- *If $x_1 = x_2$ and $y_1 \neq y_2$, then $P + Q = \mathcal{O}$.*
- *If $P = Q$ and $y_1 \neq 0$, then $P + Q = R = (x_3, y_3)$ where*

$$x_3 = m^2 - 2x_1, \quad y_3 = m(x_1 - x_3) - y_1, \quad m = \frac{3x_1^2 + A}{2y_1}.$$

- *If $P = Q$ and $y_1 = 0$, then $P + Q = \mathcal{O}$.*

*Moreover, we define $P + \mathcal{O} = P$ for all $P \in E(\mathbb{F})$ (including $P = \mathcal{O}$).*

**Lemma 1.3.** *The set of points $E(\mathbb{F})$ equipped with the addition defined in 1.2 forms an abelian group with $\mathcal{O}$ as the identity element and $(x, -y)$ being an inverse of $(x, y)$.*

*Proof.* Inverse and neutral elements are given by definition. Commutativity can be verified from the definition as well. The only non-trivial part is the associativity. It can be proved directly using the formulae and distinguishing between numerous cases. This approach is illustrated in Sutherland [2017], Section 2.1.2. A different, more theoretical approach uses algebraic geometry, see Washington [2008], Section 4.1. □

*Note.* We will follow the usual group-theoretic convention and use $E(\mathbb{F})$ to denote both the set and the group.

# 2. Basic Theory

## 2.1 Torsion points

When studying properties of an elliptic curve $E(\mathbb{F})$ it is often useful to look at the points with coordinates in the algebraic closure $\overline{\mathbb{F}}$.

**Definition 2.1.** *Let $E(\mathbb{F})$ be an elliptic curve defined by $y^2 = x^3 + Ax + B$, with $A, B \in \mathbb{F}$. We define*

$$E(\overline{\mathbb{F}}) = \{\mathcal{O}\} \cup \left\{(x, y) \in \overline{\mathbb{F}} \times \overline{\mathbb{F}} \mid y^2 = x^3 + Ax + B\right\}.$$

*Note.* The set $E(\overline{\mathbb{F}})$ forms a group. This follows immediately from Lemma 1.3 by noting that $A, B \in \overline{\mathbb{F}}$.

As in every additive group, for a point $P$ on an elliptic curve $E(\mathbb{F})$ we define the expression $kP$ to be the sum

$$\underbrace{P + P + P + ... + P}_{k \text{ times}}.$$

We can now look at an important concept of $n$-torsion points. These are the points in $E(\overline{\mathbb{F}})$ of order $m$ such that $m \mid n$.

**Definition 2.2.** *Let $E(\mathbb{F})$ be an elliptic curve and let $n \in \mathbb{N}$. We define the set of $n$-torsion points of $E(\mathbb{F})$ as*

$$E[n] = \left\{P \in E(\overline{\mathbb{F}}) \mid nP = \mathcal{O}\right\}.$$

*Note.* One can easily see that the set of $n$-torsion points $E[n]$ forms a subgroup of $E(\overline{\mathbb{F}})$.

*Example.* Let us take a look at $E[2]$. The polynomial $x^3 + Ax + B$ has exactly three roots in $\overline{\mathbb{F}}$. Since we are considering only non-singular elliptic curves, all the roots are distinct. Denote the roots $r_1, r_2, r_3 \in \overline{\mathbb{F}}$. We can write

$$y^2 = (x - r_1)(x - r_2)(x - r_3).$$

For each root $r_i$ we have a point $(r_i, 0) \in E(\overline{\mathbb{F}})$. Since $-(r_i, 0) = (r_i, 0)$ we have

$$2(r_i, 0) = (r_i, 0) + (r_i, 0) = (r_i, 0) - (r_i, 0) = \mathcal{O}, \quad i \in \{1, 2, 3\}.$$

These and the point at infinity are the only points with this property. Therefore

$$E[2] = \{\mathcal{O}, (r_1, 0), (r_2, 0), (r_3, 0)\}.$$

Since there is no point of order 4, $E[2]$ is isomorphic to $\mathbb{Z}_2 \oplus \mathbb{Z}_2$.

The following theorem provides an insight into the structure of the group of $n$-torsion points. We will regard it as fact for now. It can be proved using division polynomials defined in the next section, see Washington [2008] (chapters 3.1 and 3.2).

**Theorem 2.3.** *Let $E(\mathbb{F})$ be an elliptic curve and let $n \in \mathbb{N}$. If $char(\mathbb{F})$ does not divide $n$, including $char(\mathbb{F}) = 0$, then*

$$E[n] \simeq \mathbb{Z}_n \oplus \mathbb{Z}_n \ .$$

*If $char(\mathbb{F})$ is equal to $p > 0$ and $p \mid n$ write $n = p^r n'$ with $p \nmid n'$. Then*

$$E[n] \simeq \mathbb{Z}_{n'} \oplus \mathbb{Z}_{n'} \quad or \quad E[n] \simeq \mathbb{Z}_n \oplus \mathbb{Z}_{n'} \ .$$

## 2.2 Division polynomials

In order to compute $nP$ for $P \in E(\mathbb{F})$, $n \in \mathbb{N}$, we need to apply the addition formula $n$-times. Computationally, this can be sped up using the double-and-add method. However, there seems to be no way to determine the coordinates of $nP$ directly. In order to achieve this (at least for theoretical purposes) we make use of division polynomials.

**Definition 2.4.** *Let $x, y, A, B$ be variables. We define the $m$-th **division polynomial** $\psi_m \in \mathbb{Z}[x, y, A, B]$, $m \in \mathbb{N}_0$, recursively as follows:*

$$\begin{aligned}
\psi_0 &= 0, \\
\psi_1 &= 1, \\
\psi_2 &= 2y, \\
\psi_3 &= 3x^4 + 6Ax^2 + 12Bx - A^2, \\
\psi_4 &= 4y(x^6 + 5Ax^4 + 20Bx^3 - 5A^2x^2 - 4ABx - 8B^2 - A^3), \\
\psi_{2m+1} &= \psi_{m+2}\psi_m^3 - \psi_{m-1}\psi_{m+1}^3, \quad \text{for } m \geq 2, \\
\psi_{2m} &= (2y)^{-1}(\psi_m)(\psi_{m+2}\psi_{m-1}^2 - \psi_{m-2}\psi_{m+1}^2), \quad \text{for } m \geq 3.
\end{aligned}$$

*Note.* From the definition it may not be obvious that $\psi_{2m}$ is a polynomial. Following lemma will, among other things, show that this is indeed the case.

**Lemma 2.5.** *Let $n \in \mathbb{N}$ and let $\psi_n$ be defined as above. Then we have:*

1. *If $n$ is odd, $\psi_n$ is a polynomial in $\mathbb{Z}[x, y^2, A, B]$.*
2. *If $n$ is even, $\psi_n$ is a polynomial in $2y\mathbb{Z}[x, y^2, A, B]$.*

*Proof.* Proceed by induction on $n$. Both assumptions hold for $n \leq 4$. Assume $n > 4$. In both cases we will assume that the induction assumptions hold for all $k < n, k \in \mathbb{N}$.

First, consider the case $n = 2m + 1$ for some $m \geq 2$. Then $m - 1 > 0$ and $m + 2 < 2m + 1$ so all of the polynomials in the definition of $\psi_{2m+1}$ satisfy the induction assumptions. If $m$ is even, then $\psi_m, \psi_{m+2} \in 2y\mathbb{Z}[x, y^2, A, B]$ and consequently $\psi_{m+2}\psi_m^3 \in \mathbb{Z}[x, y^2, A, B]$. Since $m + 1$ and $m - 1$ are odd, we have $\psi_{m-1}\psi_{m+1}^3 \in Z[x, y^2, A, B]$. Together, it follows that $\psi_{2m+1} = \psi_{m+2}\psi_m^3 - \psi_{m-1}\psi_{m+1}^3 \in Z[x, y^2, A, B]$. For $m$ odd the proof is similar.

Next, consider $n = 2m$ for some $m \geq 3$. Then $m - 2 > 0$ and $m + 2 < 2m$, so all of the polynomials in the definition of $\psi_{2m}$ satisfy the induction assumptions. If $m$ is even, then $\psi_{m-2}, \psi_m, \psi_{m+2} \in 2y\mathbb{Z}[x, y^2, A, B]$ so $(\psi_m)(\psi_{m+2}\psi_{m-1}^2 - \psi_{m-2}\psi_{m+1}^2) \in (2y)^2 Z[x, y^2, A, B]$. Multiplying by $(2y)^{-1}$ leaves us with $\psi_{2m} = (2y)^{-1}(\psi_m)(\psi_{m+2}\psi_{m-1}^2 - \psi_{m-2}\psi_{m+1}^2) \in 2y\mathbb{Z}[x, y^2, A, B]$. For $m$ odd the proof is similar. □

In order to simplify the final formula for $nP$ we define two additional polynomials.

**Definition 2.6.** *Let $n \geq 2, n \in \mathbb{N}$. We define*

$$\phi_n = x\psi_n^2 - \psi_{n+1}\psi_{n-1},$$
$$\omega_n = (4y)^{-1}(\psi_{n+2}\psi_{n-1}^2 - \psi_{n-2}\psi_{n+1}^2).$$

**Lemma 2.7.** *Let $n \geq 2, n \in \mathbb{N}$. Then we have*

1. *$\phi_n \in \mathbb{Z}[x, y^2, A, B]$.*
2. *If $n$ is odd, then $\omega_n \in y\mathbb{Z}[x, y^2, A, B]$, otherwise $\omega_n \in \frac{1}{2}\mathbb{Z}[x, y^2, A, B]$.*

*Proof.* We will prove each point separately.

1. If $n$ is odd, from Lemma 2.5 we have $\psi_n \in \mathbb{Z}[x, y^2, A, B]$ and $\psi_{n+1}\psi_{n-1} \in (2y)^2 Z[x, y^2, A, B] \subset Z[x, y^2, A, B]$. Together it follows that

   $$\phi_n = x\psi_n^2 - \psi_{n+1}\psi_{n-1} \in \mathbb{Z}[x, y^2, A, B].$$

   If $n$ is even, the proof is similar.

2. If $n$ is odd, Lemma 2.5 gives $\psi_{n-1}^2, \psi_{n+1}^2 \in 4y^2\mathbb{Z}[x, y^2, A, B]$. It follows that $\psi_{n+2}\psi_{n-1}^2 - \psi_{n-2}\psi_{n+1}^2 \in 4y^2\mathbb{Z}[x, y^2, A, B]$ and multiplying by $(4y)^{-1}$ yields

   $$\omega_n = (4y)^{-1}(\psi_{n+2}\psi_{n-1}^2 - \psi_{n-2}\psi_{n+1}^2) \in y\mathbb{Z}[x, y^2, A, B].$$

   If $n$ is even, the proof is again very similar.

□

*Note.* It can be shown that $\omega_n \in \mathbb{Z}[x, y^2, A, B]$ for $n$ even. However, it is a bit more technical and we omit the proof. For details see Washington [2008], Section 3.2.

The following result provides us with an explicit formula for $nP$. In order to prove it, a more advanced theory is required, see Washington [2008], Section 9.5.

**Theorem 2.8.** *Let $E(\mathbb{F})$ be an elliptic curve defined by $y^2 = x^3 + Ax + B$. For $P = (x, y) \in E(\overline{\mathbb{F}})$ and $n \in \mathbb{N}$ we have*

$$nP = \left(\frac{\phi_n(x)}{\psi_n^2(x)}, \frac{\omega_n(x, y)}{\psi_n^3(x, y)}\right).$$

## 2.3 Endomorphisms

This section aims to lay the definitions necessary to state the Theorem 2.13 which is one of the most important building blocks of the Schoof's algorithm.

**Definition 2.9.** ***Endomorphism*** *of an elliptic curve $E(\mathbb{F})$ is a homomorphism $\alpha\colon E(\overline{\mathbb{F}}) \to E(\overline{\mathbb{F}})$, that is given by rational functions, i.e.*

$$\alpha(x, y) = \Big(R_1(x, y), R_2(x, y)\Big), \quad \text{where } R_1, R_2 \in \mathbb{F}(x, y).$$

*Note.* It can happen that the rational functions $R_1, R_2$ are not defined at a point. We will treat this case after the following lemma.

**Lemma 2.10.** *Every endomorphism $\alpha$ of an elliptic curve $E(\mathbb{F})$ can be written as $\alpha(x,y) = \big(r_1(x), y\, r_2(x)\big)$ where $r_1, r_2 \in \mathbb{F}(x)$.*

*Proof.* The elliptic curve $E(\mathbb{F})$ is defined by $y^2 = x^3 + Ax + B$. This relation can be used to replace all even powers of $y$ by a polynomial in $x$. So for a rational function $R(x,y)$ we get

$$R(x,y) = \frac{p_1(x) + yp_2(x)}{p_3(x) + yp_4(x)},$$

where $p_i(x)$ are polynomials in $x$ alone. Multiplying by $p_3(x) - yp_4(x)$ we get

$$\frac{p_1(x)p_3(x) + yp_2(x)p_3(x) - yp_1(x)p_4(x) - y^2p_2(x)p_4(x)}{p_3^2(x) - y^2p_4^2(x)} = \frac{q_1(x) + yq_2(x)}{q_3(x)}$$

for some polynomials $q_i(x)$.

Now, we will use the fact that $\alpha$ is an endomorphism given by $\alpha(x,y) = (R_1(x,y), R_2(x,y))$. We have

$$\alpha(x,-y) = \alpha\left(-(x,y)\right) = -\alpha(x,y),$$

which means that

$$(R_1(x,-y), R_2(x,-y)) = \left(R_1(x,y), -R_2(x,y)\right).$$

Thus for $R_1$ we have

$$R_1(x,y) = \frac{q_1(x) + yq_2(x)}{q_3(x)} = \frac{q_1(x) - yq_2(x)}{q_3(x)} \implies yq_2(x) = 0.$$

It must be the case that $q_2(x) = 0$. Consequently, we have

$$R_1(x,y) = \frac{q_1(x)}{q_3(x)} = r_1(x)$$

for a rational function $r_1(x)$, so $R_1$ does not depend on $y$.
On the other hand, for $R_2$ we have

$$R_2(x,-y) = \frac{s_1(x) - ys_2(x)}{s_3(x)} = \frac{-s_1(x) - ys_2(x)}{s_3(x)} \implies s_1(x) = 0.$$

It follows that

$$R_2(x,y) = yr_2(x)$$

for a rational function $r_2(x)$. □

Let $\alpha(x,y) = (r_1(x), yr_2(x))$ be an endomorphism. What happens when $r_1(x)$ or $r_2(x)$ is not defined at a point? Write $r_1(x) = p_1(x)/q_1(x)$ and $r_2(x) = p_2(x)/q_2(x)$ for some polynomials $p_1(x), q_1(x), p_2(x), q_2(x) \in \mathbb{F}[x]$. We can assume that $p_i$ and $q_i$ are relatively prime for $i = 1, 2$.

If $q_1(x_0) = 0$ for some $x_0 \in \overline{\mathbb{F}}$, we define

$$\alpha(x_0, y_0) = \mathcal{O}.$$

Conversely, suppose $q_2(x_0) = 0$ for some $x_0 \in \overline{\mathbb{F}}$. The relation $y^2 = x^3+Ax+B$ holds for all the points $(x, y) \in E(\overline{\mathbb{F}})$. Consequently, it must hold for all the points of the form $(r_1(x), yr_2(x))$. Therefore, we have

$$\left(\frac{p_1(x)}{q_1(x)}\right)^3 + A\frac{p_1(x)}{q_1(x)} + B = (x^3 + Ax + B)\left(\frac{p_2(x)}{q_2(x)}\right)^2.$$

The polynomial $x^3 + Ax + B$ has no double roots (we are working with non-singular curves) and $q_2^2(x)$ has only double roots. Since $p_2$ and $q_2$ are relatively prime, there must be at least one factor of the form $(x - x_0)$ remaining in the denominator. Therefore, the right-hand side is not defined at $x_0$. It follows that the left-hand side is not defined at $x_0$ either. Thus, we have $q_1(x_0) = 0$.

*Note.* The endomorphism $\alpha(P) = \mathcal{O}$, $\forall P \in E(\overline{\mathbb{F}})$, will be denoted as 0.

**Definition 2.11.** ***Frobenius map*** $\phi_q$ *on an elliptic curve* $E(\mathbb{F}_q)$ *is defined as*

$$\phi_q(x, y) = (x^q, y^q)$$

*for* $(x, y) \in E(\overline{\mathbb{F}_q})$. *For* $\mathcal{O}$ *we define* $\phi_q(\mathcal{O}) = \mathcal{O}$.

**Lemma 2.12.** *Let* $E(\mathbb{F}_q)$ *be an elliptic curve and let* $(x, y) \in E(\overline{\mathbb{F}_q})$. *Then*

1. $\phi_q(x, y) \in E(\overline{\mathbb{F}_q})$.
2. $(x, y) \in E(\mathbb{F}_q)$ *if and only if* $\phi_q(x, y) = (x, y)$.

*Proof.* For (1) we work with the equation

$$y^2 = x^3 + Ax + B, \tag{2.1}$$

which by definition holds for all $(x, y) \in E(\overline{\mathbb{F}_q})$. For $a, b \in \overline{\mathbb{F}_q}$ we have $(a + b)^q = a^q + b^q$. Raising both sides of (2.1) to the power of $q$ we have

$$\left(y^2\right)^q = \left(x^3 + Ax + B\right)^q,$$

$$(y^q)^2 = (x^q)^3 + A^q x^q + B^q.$$

Since $A, B \in \mathbb{F}_q$ we have $A^q = A$ and $B^q = B$. We end up with

$$(y^q)^2 = (x^q)^3 + Ax^q + B,$$

which means that the point $(x^q, y^q)$ satisfies the equation (2.1). In other words, $(x^q, y^q) \in E(\overline{\mathbb{F}_q})$.

For (2), recall that for $a \in \overline{\mathbb{F}_q}$ we have $a \in \mathbb{F}_q$ if and only if $a^q = a$. Therefore, for $(x, y) \in E(\overline{\mathbb{F}_q})$ we have $(x, y) \in E(\mathbb{F}_q)$ if and only if $(x^q, y^q) = (x, y)$. □

*Note.* Frobenius map is an endomorphism of $E(\mathbb{F}_q)$. Clearly, it is given by rational functions (in fact, polynomials). The fact that it is a homomorphism, i.e.

$$\phi_q(P + Q) = \phi_q(P) + \phi_q(Q),$$

can be proved by case analysis of $P$ and $Q$. For details, see Washington [2008], Section 2.9.

Endomorphisms of the group $E(\overline{\mathbb{F}_q})$ actually form a ring with respect to the operations of addition and composition. This leads to many interesting results one of which is given below. The proof can be found in Washington [2008], Section 4.2 or alternatively in Sutherland [2017], Lecture 7.

**Theorem 2.13.** *Let $E(\mathbb{F}_q)$ be an elliptic curve. Let $a = q + 1 - \#E(\mathbb{F}_q)$ where $\#E(\mathbb{F}_q)$ denotes the order of the group $E(\mathbb{F}_q)$. Then*

$$\phi_q^2 - a\phi_q + q = 0$$

*and $a$ is the unique integer $k$ such that*

$$\phi_q^2 - k\phi_q + q = 0.$$

*Note.* The expression $\phi_q^2 - a\phi_q + q = 0$ is a shorthand for

$$\phi_q\left(\phi_q(x, y)\right) - a\phi_q(x, y) + q(x, y) = \mathcal{O},$$

where multiplication by integers $a$ and $q$ is interpreted as iterative addition.

Following theorem, originally conjectured by Emil Artin, was proved by Hasse in 1933. It restricts the number $\#E(\mathbb{F}_q)$ to differ from $q+1$ by at most $2\sqrt{q}$. The original proof can be found in Hasse [1936].

**Theorem 2.14** (Hasse)**.** *Let $E(\mathbb{F}_q)$ be an elliptic curve where $q = p^k$ for a prime $p$ and $k \in \mathbb{N}$. Let $\#E(\mathbb{F}_q)$ denote the order of $E(\mathbb{F}_q)$. Then*

$$|q + 1 - \#E(\mathbb{F}_q)| \leq 2\sqrt{q}.$$

# 3. Schoof's algorithm

Schoof's algorithm was the first deterministic polynomial-time algorithm to compute the order of $E(\mathbb{F}_q)$. It has been introduced by René Schoof in 1985, see Schoof [1985]. Since then, it has been improved by N. Elkies and A. O. L. Atkin to Schoof-Elkies-Atkin algorithm which is currently the fastest algorithm for computing $\#E(\mathbb{F}_q)$. For $q \sim 2^{256}$, a size suitable for modern elliptic curve cryptography, it takes only a few seconds to compute $\#E(\mathbb{F}_q)$.

## 3.1 Overview

For the purpose of this algorithm we will consider $q = p^k$ for a prime $p > 3$. From Hasse's theorem 2.14, we have

$$\#E(\mathbb{F}_q) = q + 1 - a, \quad \text{with } |a| \leq 2\sqrt{q}.$$

The idea of Schoof's algorithm is to compute $a \bmod l$ for many small primes $l$. Once the product $\prod_{l \in S} l$ is larger than $4\sqrt{q}$, $a$ can be uniquely determined using Chinese remainder theorem.

Here is an overview of the algorithm:

1: Set $S$ to be a set of primes not dividing $q$ such that $\prod_{l \in S} l \geq 4\sqrt{q}$
2: **for all** $l$ in $S$ **do**
3: Find $a_l \in \mathbb{Z}$ such that $a_l = a \bmod l$
4: **end for**
5: Find $a \in \mathbb{Z}, |a| \leq 2\sqrt{q}$ such that $a \equiv a_l \bmod l$ for each $l \in S$
6: **return** $q + 1 - a$

The running time of the algorithm is dominated by computations in step 3 - computing $a \bmod l$. This step will be analyzed in detail in the next section. Now, we present the corresponding implementation of the algorithm using SageMath.

```
def Schoof(E,q):
    """ compute the order of the elliptic curve E(F_q)
    given by y^2 = x^3 + A*x + B """
    # create S = {2,3,5...} set of primes such that
    # their product is greater than 4*sqrt(q)
    M = 1
    S = []
    for p in Primes():
        if gcd(q,p)>1: # skip p = char(F_q)
            continue
        S.append(p)
        M = M*p
        if M>4*sqrt(q):
            break
    a_mod = {} # dictionary to store the values a mod l
    # compute and store (a mod l) for all primes l in S
```

```
    for l in S:
        a_mod[l] = Frobenius_mod(q, l, E)
    # using the Chinese Remainder Theorem, find (a mod M)
    a = CRT(a_mod.values(), a_mod.keys())
    # if a is too large, subtract M, abs(a) has to be
    # at most 2*sqrt(q)
    if a>M/2:
        a = a - M
    return q + 1 - a
```

## 3.2 Trace of Frobenius map

There are two cases to consider when computing $a \bmod l$.

Case $l = 2$: Determine whether $x^3 + Ax + B$ has a root in $\mathbb{F}_q$. If there is a root, say $r_1$, then $(r_1, 0) + (r_1, 0) = \mathcal{O}$, so $(r_1, 0)$ has order 2. From Lagrange's theorem we have $2 \mid q + 1 - a$. Since $q$ is odd, it follows that $a \equiv 0 \pmod 2$.

On the other hand, if $x^3 + Ax + B$ has no root in $\mathbb{F}_q$, then there is no point of order 2 on $E(\mathbb{F}_q)$ and therefore $2 \nmid q + 1 - a$. Since $q$ is odd, it follows that $a \equiv 1 \pmod 2$.

Case $l > 2$: From the Theorem 2.13 we have

$$\phi_q^2(x, y) - a\phi_q(x, y) + q(x, y) = \mathcal{O}$$

for all $(x, y) \in E(\overline{\mathbb{F}_q})$. However, if $(x, y) \in E[l]$, then also

$$\phi_q^2(x, y) - a_l\phi_q(x, y) + q_l(x, y) = \mathcal{O} \tag{3.1}$$

holds, where $a_l = a \bmod l$ and $q_l = q \bmod l$. Since $l$ is a relatively small prime, this reduces the size of $a$ and $q$ enormously. We will proceed by computing the left-hand side of

$$(x^{q^2}, y^{q^2}) + q_l(x, y) = a_l(x^q, y^q) \tag{3.2}$$

treating the terms as elements of $End(E[l])$ with coefficients given by rational functions. We can restrict the computation to the ring $\mathbb{F}_q[x]$, regarding the variable $y$ as implicit and modifying the addition formulae accordingly. Moreover, to speed computations up, we can use the $l$-th division polynomial $\psi_l(x)$ of degree $(l^2 - 1)/2$ to reduce the degree of polynomials involved in these computations. Therefore, we will be working in the ring $\mathbb{F}_q[x]/\psi_l(x)$. This is thanks to the fact that $(x_0, y_0) \in E[l]$ if and only if $\psi_l(x_0) = 0$.

Once the left-hand side of (3.2) is computed we can search for $a_l \in \mathbb{Z}_l$ on the right-hand side, such that the relation holds.

However, there is still one caveat. When adding the elements of $End(E[l])$ using the formulae for point addition, a non-invertible denominator may emerge. This causes the algorithm to restart, replacing the $\psi_l(x)$ with a factor that has been found. As a consequence, the algorithm runs faster, reducing the elements by a polynomial with a lower degree. This property of certain primes $l$ can be systematically used to further reduce the complexity and forms the basic idea behind the improvements found by Elkies and Atkin. For more information, see Sutherland [2017], Lectures 9 and 20.

Following, the computation of $a \bmod l$ is implemented.

```
def Frobenius_mod(q,l,E):
    """ compute trace of Frobenius map phi_q on E modulo
    prime l """
    global g
    Fq.<z> = PolynomialRing(E.base_ring()) # Fq[z]
    A = E.a4(); B = E.a6()
    # case l = 2
    if l==2:
        if (z^3 + A*z + B).is_irreducible():
            # z^3 + A*z + B has no roots in Fq
            return 1
        else:
            # z^3 + A*z + B has a root in Fq
            return 0
    # case l>2
    h = E.division_polynomial(l,z)
    q_l = q % l
    j = 1
    while True:
        R.<x> = Fq.quotient(ideal(h)) # ring F[x]/h(x)
        # compute the left hand side of of
        # phi_q^2 + q_l = a_l*phi_q
        f = x^3 + A*x + B
        phi = ( x^q , f^((q-1)//2) )
        phi_squared = ( phi[0]^q , phi[1]^(q+1) )
        identity = (x, R(1))
        try: # non-invertible elements may emerge
            # q_l as an element of End(E[l])
            Q = int_times_endm(q_l, identity, f, A)
            # Left-hand side of phi_q^2 + q_l = a_l*phi_q
            L = add_endm( phi_squared, Q, f, A)
        except ZeroDivisionError:
            # restart this iteration of "while True" loop
            # with a new h
            h = gcd(h, g.lift())
            continue
        if L == 0: # we have 0 = a_l*phi_q, a_l must be 0
            return 0
        # for j in {0,1,..,(l-1)/2} compute Rj = j*phi
        # until there is a match L = Rj or L = -Rj
        Rj = 0
        while j <= (l-1)/2:
            try: # non-invertible elements may emerge
                Rj = add_endm(Rj, phi, f, A)
            except ZeroDivisionError:
                # restart this iteration of "while True"
                # loop with a new h
                h = gcd(h, g.lift())
```

```
            break
        if L == Rj:
            return j
        if L[0] == Rj[0] and L[1] == -Rj[1]:
            return -j
        j = j + 1
    assert False
```

## 3.3 Adding endomorphisms

For completeness, we present the modified algorithms for addition of elements of $End(E[l])$ and multiplication by an integer.

```
def add_endm(P, Q, f, A):
    """ compute the sum of endomorphisms given by P and Q,
    using appropriate formula """
    global g
    # if either one is 0, return the other
    if P==0:
        return Q
    if Q==0:
        return P
    # if P = -Q, the sum is 0
    if P[0] == Q[0] and P[1] == -Q[1]:
        return 0
    # if P = Q, use the "tangent" formula
    if P==Q:
        try: # denominator might not be invertible
            m = (3*P[0]^2 + A) / (2*P[1]*f)
        except ZeroDivisionError:
            g = 2*P[1]*f
            raise
        R = [0,0]
        R[0] = m^2*f - 2*P[0]
        R[1] = m*(P[0]-R[0]) - P[1]
        return ( R[0] , R[1] )
    # if none of the above, use the "line through P and Q"
    # formula
    try: # denominator might not be invertible
        m = (Q[1] - P[1]) / (Q[0] - P[0])
    except ZeroDivisionError:
        g = Q[0] - P[0]
        raise
    R = [0,0]
    R[0] = f*m^2 - P[0] - Q[0]
    R[1] = m*(P[0] - R[0]) - P[1]
    return ( R[0] , R[1])
```

To compute $nP$, for an endomorphism given by coordinates of $P$, we use the double-and-add method.

```
def int_times_endm(n, P, f, A):
    """ compute n*P for endomorphism given by P, using
    double and add method """
    global g
    if n==0:
        return 0
    if n==1:
        return P
    nbits = n.digits(2)
    R = P
    for i in reversed(range(len(nbits) - 1)):
        R = add_endm(R, R, f, A)
        if nbits[i]==1:
            R = add_endm(R, P, f, A)
    return R
```

# Conclusion

This thesis aimed to provide a basic understanding of the theory of elliptic curves necessary to understand the Schoof's algorithm for counting points on elliptic curves over finite fields. Some of the results have been presented as facts for the sake of conciseness and readability. We consider the main contribution to be the implementation of the Schoof's algorithm.

# Bibliography

Helmut Hasse. Zur theorie der abstrakten elliptischen Funktionenkörper. I, II & III. *Crelle's Journal*, 1936.

René Schoof. Elliptic curves over finite fields and the computation of square roots mod p. *Mathemtics of Computation*, 44, 1985.

Joseph H. Silverman. *The Arithmetic of Elliptic Curves*. Second edition. Springer, 2009. ISBN 978-0-387-09494-6.

Andrew Sutherland. *18.783 - Elliptic Curves - lecture notes from MIT*. math.mit.edu/classes/18.783/2017/lectures.html, 2017.

Lawrence C. Washington. *Elliptic curves - number theory and cryptography*. Second edition. Taylor & Francis Group, LLC, 2008. ISBN 978-1-4200-7146-7.