Re-discovering ECM

Alessandro Colombo

School of Computer and Communication Sciences

Semester Project

May 2023

Responsible
Prof. Serge Vaudenay
EPFL / LASEC

Supervisor
Dr. Tako Boris Fouotsa
EPFL / LASEC

Abstract

In this report, we explore the Integer Factorization Problem by studying some special purpose factorization algorithms, giving particular emphasis to the Elliptic Curve Factorization Method (ECM). Furthermore, we propose some variants of ECM that use specific curves over $\mathbb{Q}$ having special endomorphism rings. These curves are said to have *complex multiplication* and, under certain conditions, can be generated such that they have a predefined order. We first design a $p+1$ version of ECM by replacing the curves used in the algorithm with potential supersingular curves, and we experimentally compare this method with the Williams $p+1$ algorithm. Next, we propose a second variant of ECM that uses anomalous curves. This method leads to a *polynomial* time algorithm that factors a special class of RSA moduli having a prime divisor of the form $p = Dm(m+1) + \frac{D+1}{4}$ for a *small* integer $D \equiv 3 \pmod 4$ and $m > 0$. Even though honestly generated RSA moduli fall into this class with negligible probability, the algorithm exposes a class of vulnerable moduli. For instance, a malicious entity could corrupt the key generation algorithm to generate those vulnerable RSA moduli and later be able to retrieve users' secret keys efficiently.

Chapter 1

General Introduction

In this chapter, we introduce the content of the report. We start by expressing the motivations of the project (Section 1.1) and introducing its main topics (Section 1.2). Then, we list the contributions of our work (Section 1.3), and we conclude with an outline of the following chapters (Section 1.4).

1.1 Motivation

The goal of this project was to study the Elliptic Curve Factorization Method (ECM) developed by H.Lenstra [1] in 1987 and investigate some possible improvements. Such improvements are based on the research of Dr.Boris Fouotsa [2] and involve the usage of curves of specific order inside the ECM algorithm. When the number to be factored satisfies certain conditions, this approach may result in considerable performance improvements, which we will try to investigate in the rest of the report.

1.2 Introduction

The Integer Factoring Problem is, together with the Discrete Logarithm Problem, one of the biggest problems of classic Public Key Cryptography (PKC). The security of the widely deployed Rivest–Shamir–Adleman (RSA) cryptosystem [3] is based on the hardness assumption of this problem, which can be instantiated in the following way: given a *large* semi-prime $n = pq$, find a prime factor p of n. The semi-prime n is often called an RSA *modulus*. Nowadays, typical lengths for RSA moduli are 2048 or 4092, even though these lengths are likely to increase in the future, due to advances in cryptanalysis and hardware performance.

For centuries, many famous mathematicians have been working on finding efficient solutions to the Integer Factorization Problem, especially after the introduction of RSA. However, no classical polynomial time algorithm

to solve the generic instance of the problem has ever been discovered. Factorization methods can be split into two categories: the general purpose algorithms (e.g., the Quadratic Sieve, the Number Field Sieve) whose runtime depends exclusively on the length of the number to factor, and special purpose algorithm (e.g., Pollard's rho, Pollard's $p-1$, Williams $p+1$, ECM) whose efficiency depends on some property of the number to factor (or of one of its factors).

The ECM algorithm is a special purpose factorization algorithm published by H. Lenstra in 1987 [1], and it is inspired by Pollard's $p-1$ algorithm [4]. To factor an integer $n = pq$, Pollard's algorithm fixes a bound $B \in \mathbb{Z}$, randomly picks an integer $x \in \mathbb{Z}_n \setminus \{0, 1, n-1\}$, and computes $gcd(x^m, n)$, with $m = \prod p_i^{\lfloor \log_{p_i} B \rfloor}$ for all primes $p_i \leq B$. When $p-1$ is B-smooth and $q-1$ is not, this method succeeds in factoring n with complexity $O(B)$. The algorithm exploits a consequence of Fermat's Little Theorem. Namely, for every prime p, the order of the multiplicative group $\mathbb{Z}_p^\times$ is exactly $p-1$.

Instead of working with the multiplicative group $\mathbb{Z}_p^\times$, Lenstra's idea was to use the additive group of points of an elliptic curve E defined over a finite field $\mathbb{F}_p$. In this way, the algorithm succeeds when the *group order* of E is B-smooth. In particular, one picks a random elliptic curve E defined over $\mathbb{Z}_n$ and a point $P \in E(\mathbb{Z}_n)$ and then computes $[m]P$ (with m defined as before). If $\#E(\mathbb{F}_p)$ is B-smooth and $\#E(\mathbb{F}_q)$ is not, during the computation of $[m]P$, a divisor D will be invertible over $\mathbb{F}_q$ but not over $\mathbb{F}_p$, i.e., $[m]P = O$ over $\mathbb{F}_p$ and $[m]P \neq O$ over $\mathbb{F}_q$. Hence, computing $gcd(D, n)$ yields a non-trivial factor of n. The advantage here is that, while the order of $\mathbb{Z}_p^\times$ is fixed to $p-1$, the order of a random elliptic curve E over $\mathbb{F}_p$ varies between Hasse's interval $\#E(\mathbb{F}_p) = p + 1 - t$ with $|t| < 2\sqrt{p}$. Hence, one can keep sampling random elliptic curves over $\mathbb{Z}_n$ until one of them has B-smooth order and the above algorithm succeeds. Due to Hasse's bound, the more n is unbalanced (e.g., p is smaller than q), the faster ECM will succeed, as $\#E(\mathbb{F}_p) \simeq p$ has more chances of being B-smooth. However, as honestly generated RSA moduli should have balanced prime divisors (i.e., p and q should be of the same size), this property has limited impact in the case of RSA factorization.

Supersingular curves are elliptic curves having order $p+1$ when defined over a prime field $\mathbb{F}_p$. Assume we are attempting to factor an RSA modulus $n = pq$ such that $p+1$ is B-smooth, and that we are given a supersingular elliptic curve E over $\mathbb{F}_p$ with a point $P \in E(\mathbb{F}_p)$. As $p+1$ is B-smooth, if we compute $[m]P$ over $\mathbb{Z}_n$ we have $[m]P = O$ over $\mathbb{F}_p$ and $[m]P \neq O$ over $\mathbb{F}_q$. Hence, we will succeed in factoring n (unless E is supersingular also over $\mathbb{F}_q$ and $q+1$ is B-smooth). We observe that, when ECM is executed over supersingular curves, it can be seen as a $p+1$ factoring method with asymptotic complexity $O(B)$.

Anomalous curves are elliptic curves defined over a prime field $\mathbb{F}_p$ that have cardinality p. Suppose we are given an RSA modulus n, an anomalous curve E defined over $\mathbb{F}_p$ such that $p|n$, and a point $P \in E(\mathbb{F}_p)$. As $\#E(\mathbb{F}_p)$ divides n, if we compute $[n]P$ over $\mathbb{Z}_n$ we have $[n]P = O$ over $\mathbb{F}_p$ and $[n]P \neq O$ over $\mathbb{F}_q$. Hence, this algorithm yields a non-trivial factor of n in *polynomial* time. For the algorithm to work, we need n to admit a prime factor of form $4p = D(2m+1)^2+1$, where $m > 0$, and $-D$ is the discriminant of an order in an imaginary quadratic field. This is in fact the case for every odd prime p. For example, one can first observe that for an odd prime p, we have $4p - 1 \equiv 3 \pmod 4$. Moreover, every odd integer $y = 2m + 1$ is such that $y^2 \equiv 1 \pmod 8$, and all the negative integers $-D \equiv 1 \pmod 4$ are discriminants by definition (notice that, for all of them, $Dy^2 \equiv -1 \pmod 4$). Hence, the condition $4p-1 \equiv Dy^2$ is always verified. However, for the algorithm to be efficient we need D to be *small*, which is very unlikely to happen in a random RSA modulus. Indeed, the probability that an integer $y \simeq p$ is a perfect square is negligible. Hence, when p is of cryptographic size and D is small (e.g., $D < 1000$), the probability that $\frac{4p-1}{D}$ is a perfect square is also negligible.

1.3 Contributions

The project was structured in two phases: the first phase was aimed at implementing a $p + 1$ factorization method based on supersingular elliptic curves and comparing it to a classic $p + 1$ method (i.e, Williams $p + 1$). In the second phase, we designed and implemented an algorithm based on anomalous elliptic curves, that factors very specific classes of RSA moduli in *polynomial* time. For the algorithm to work, we need a prime factor p of n to be of the form $p = Dm(m+1) + \frac{D+1}{4}$ for small D, hence the anomalous method does not introduce an efficient solution to the RSA factorization problem. However, it exposes a class of vulnerable RSA moduli that can be factored very efficiently. We implemented a proof of concept of the Williams $p + 1$ algorithm and of the ECM algorithm in both the anomalous and the supersingular case using SageMath [1]. For more efficiency, we also developed a C implementation of ECM using the GMP multiprecision library[2]. More details can be found in Section 4.4. During the last weeks of the project, we discovered that similar versions of our anomalous approach have already been proposed by [5, 6].

[1] `https://www.sagemath.org`
[2] `https://gmplib.org`

1.4 Outline

In Chapter 2 we introduce the Pollard $p-1$ and Williams $p+1$ algorithm, in Chapter 3 we provide some useful background on elliptic curves and complex multiplication, including methods for generating curves of specific order. Then, in Chapter 4 we describe ECM, its variants with supersingular and anomalous curves, and finally we discuss the implementation. We conclude the report in Chapter 5 with some final considerations on the project.

Chapter 2

Preliminaries

In this chapter, we will present two classical factorization methods, namely Pollard $p-1$ (Section 2.1), and William $p+1$ (Section 2.2). Some of the basic ideas developed in these algorithms will be useful in the following chapters, where we will discuss the ECM.

2.1 Pollard p-1

Pollard $p-1$ is a factorization method published by J.M.Pollard in 1974 [4]. The main idea behind the algorithm is to using Fermat's Little Theorem for factoring composite numbers. Indeed, from Fermat's Little Theorem, we know that if p is prime, then $\forall a \in \mathbb{Z}_p^\times$ we have:

$$a^{(p-1)} \equiv 1 \ (mod\ p) \tag{2.1}$$

and, consequently:

$$a^{(p-1)} - 1 \equiv 0 \ (mod\ p). \tag{2.2}$$

Suppose now that we are interested in factoring an RSA modulus $n = pq$, with p, q primes. As we already mentioned, Pollard's idea consists in using Fermat's Theorem to find a prime divisor of n. Of course, as we don't know p and q beforehand, we cannot use (2.2) directly. However, lets say that p is such that $p-1$ is B-powersmooth for a *small* bound B. Then, we can rewrite it as:

$$p - 1 = (\prod_{i=1}^{k} p_i^{\alpha_i})$$

where p_i is the i-th prime dividing $p-1$, and every $p_i^{\alpha_i} \leq B$ with $\alpha_i \in \mathbb{N}$. Lets now define m as follows:

$$m = \prod_{i=1}^{k} p_i^{\beta_i} \tag{2.3}$$

where every p_i is defined as before, and each $\beta_i \in \mathbb{Z}$ is the largest integer exponent such that $p_i^{\beta_i} \leq B$. We notice that for all $i = 1, 2, ...k$, we have $\beta_i \geq \alpha_i$, meaning that $(p-1)|m$. Hence, we rewrite $m = c(p-1)$ for some $c \in \mathbb{Z}$, and thanks to (2.2) we can say that:

$$\forall a \in \mathbb{Z}_p^*,\ a^m - 1 \equiv a^{c(p-1)} - 1 \equiv 0\ (mod\ p).$$

Therefore, by taking $gcd(a^m - 1,\ n)$, one hopes to retrieve a non-trivial factor of N. This step only fails if we also have $(q-1)|m$. In such case, it is sufficient to reduce B until only $q-1$ or $p-1$ is B-powersmooth.

As a final remark, we underline that instead of fully compute m and testing the *gcd* at the end, we can split m into $m = m_1 \cdot m_2 \cdot ... \cdot m_j$ (e.g., $m_i = p_i^{\beta_i}$ for all $p_i s$) and divide the computation of a^m into $a_1 = a^{m_1}, a_2 = a_1^{m_2}, ..., a_j = a_{j-1}^{m_j}$, testing $gcd(a_i - 1, N)$ at each step until it yields a non-trivial factor. In many cases, the algorithm will stop before computing $a_j = a^m$ (assuming that $p-1$ is B-powersmooth).

The $p-1$ algorithm executed on a bound B only works when a divisor p of n is such that $p-1$ is B-powersmooth and has asymptotic complexity $O(B\ log(B)\ log^2(N))$.

2.2 Williams p+1

In this section, we will describe the William's factorization algorithm [7]. Unlike Pollard's $p-1$ method, William's method succeeds in finding a prime divisor p of N when $p+1$ is smooth (it also works when $p-1$ is smooth but has higher complexity than Pollard $p-1$).

Before delving into the algorithm, we need to define some basic properties of Lucas Functions that we will be using later.

Definition 2.2.1 (Lucas Functions) *Let $P, Q \in \mathbb{Z}$, and α, β be the zeros of $x^2 - PX + Q$. We define the Lucas functions by:*

$$U_n(P,Q) = \frac{\alpha^n - \beta^n}{\alpha - \beta}$$

$$V_n(P,Q) = \alpha^n + \beta^n$$

We also put $\Delta = (\alpha - \beta)^2 = P^2 - 4Q$.

It turns out that Lucas Functions satisfy a large number of identities. Here, we only report those required for William's method, namely:

$$\begin{cases} U_{2n} = V_n U_n \\ V_{2n} = V_n^2 - 2Q^n \end{cases} \tag{2.4}$$

$$\begin{cases} U_n(V_k(P,Q),Q^k) = U_{n,k}(P,Q)/U_k(P,Q) \\ V_n(V_k(P,Q),Q^k) = V_{nk}(P,Q) \end{cases} \tag{2.5}$$

These identities can be verified by direct substitution from Definition 2.2.1. Finally, if $gcd(N,Q) = 1$ and $P'Q \equiv P^2 - 2Q \ (mod\ N)$, then $P' \equiv \alpha/\beta + \beta/\alpha$ and $Q' \equiv \alpha/\beta \cdot \beta/\alpha \equiv 1$. Hence:

$$U_{2k}(P,Q) = PQ^{k-1}U_k(P',1) \ (mod\ N) \tag{2.6}$$

Theorem 2.2.1 (Lehmer) *if p is an odd prime, $p \nmid Q$ and the Legendre symbol $(\Delta/p) = \varepsilon$, then:*

$$U_{(p-\varepsilon)k}(P,Q) \equiv 0 \ (mod\ p)$$

$$V_{(p-\varepsilon)k}(P,Q) \equiv 2Q^{k(1-\varepsilon)/2} \ (mod\ p)$$

Now we are ready to describe Williams's algorithm. Given a bound B, suppose that:

$$p = \prod_{i=1}^{k}(q_i^{\alpha_i}) - 1 \tag{2.7}$$

where $q_i^{\alpha_i} \leq B$ for p, q_i primes (i.e., $p+1$ is B-powersmooth). If we take $m = \prod_{i=1}^{k}(q_i^{\beta_i})$ as in (2.3), we notice that $p+1 \mid m$. By Theorem 2.2.1, we see that if $(\Delta/p) = -1$ and $gcd(Q,N) = 1$, then $p \mid U_{(p+1)k}(P,Q)$, which implies $p \mid U_m(P,Q)$. Therefore, by taking $gcd(U_m(P,Q),N)$, one hopes to obtain a non-trivial factor of N.

However, William's algorithm does not compute $U_m(P,Q)$ directly, but uses a more efficient technique based on the second relation of Theorem 2.2.1. In fact, by (2.4) we know that if $p \mid U_m(P,Q)$, then $p \mid U_{2m}(P,Q)$. Thus, from (2.6), we have $p \mid U_m(P',1)$, and consequently we loose no generality by assuming $Q = 1$. With this assumption, by Theorem 2.2.1 we have:

$$V_{(p-\varepsilon)k}(P,1) \equiv 2 \ (mod\ p). \tag{2.8}$$

Therefore, we just demonstrated that if $p \mid U_m(P,1)$, then $p \mid (V_m(P,1) - 2)$.

The algorithm proceeds as follows:

Let $m = m_1 \cdot m_2 \cdot ... \cdot m_j$ as in Section 2.1, and find P_0 s.t. $gcd(P_0^2 - 4, N) = 1$. Define:

$$U_n(P) = U_n(P,1)$$
$$V_n(P) = V_n(P,1)$$

and

$$P_i \equiv V_{m_i}(P_{i-1}) \ (mod\ N) \ (i = 1,2,3,...,j).$$

By the second formula of (2.5), we notice that:

$$P_j \equiv V_m(P_0) \ (mod \ N) \tag{2.9}$$

Hence, we can calculate $gcd(P_j - 2, N)$ and see if it yields a non-trivial factor of N.

Chapter 3

Background

In this chapter, we first provide some background on elliptic curves (Section 3.1), and then we study complex multiplication (CM) and how it can be used to generate elliptic curves with predefined order (Section 3.2).

3.1 Elliptic Curves

In this section, we provide some background on elliptic curves. We start by introducing the Weierstrass equation and the short Weierstrass form (Section 3.1.1). Then, we define the group law (Section 3.1.2), and describe some useful properties (e.g., number of points, j-invariant and isomorphisms, endomorphism ring) (Section 3.1.3). We conclude by defining quadratic fields and orders (Section 3.1.4).

3.1.1 Definition and Weierstrass Equation

An elliptic curve E defined over a field K is a smooth projective curve of genus one. Here, we are mainly interested in the case where the field of definition K is a finite field $\mathbb{F}_p$ of prime characteristic p. An elliptic curve defined over K can be represented using the following Weierstrass equation:

$$E : y^2z + a_1xyz + a_3xz^2 = x^3 + a_2x^2z + a_4xz^2 + a_6z^3$$

where $(0, 1, 0)$ is the base point, and $a_1, a_2, a_3, a_4, a_6 \in K$. Such Weierstrass equation has the corresponding affine Weierstrass model:

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

obtained by substituting $x = x/z$ and $y = y/z$ when $z \neq 0$, and having one point (the base point) on the line at ∞. When the characteristic of the field $char(K)$ is different from 2 and 3, the elliptic curve is fully defined (up to isomorphisms) by the short Weierstrass equation

$$E : y^2z = x^3 + axz^2 + bz^3$$

and the corresponding affine equation:

$$E : y^2 = x^3 + ax + b \tag{3.1}$$

with $a, b \in K$ and $4a^3 + 27b^2 \neq 0$. When described by an affine equation, an elliptic curve always contains a projective point called point at infinity, that we will denote by O. We require $char(K) \neq 2, 3$ as the reduction in the short Weierstrass form involves divisions by 2 and 3 (that are not allowed when $char(K) \in \{2, 3\}$). The condition $4a^3 + 27b^2 \neq 0$ is added to ensure *non-singularity* of the curve. Just as the discriminant of a quadratic equation vanishes if the equation has repeated roots, the discriminant $\Delta = -16(4a^3 + 27b^2)$ of an elliptic curve vanishes if the curve is *singular*. Hence, *non-singularity* implies that E does not have any repeated roots (i.e., its graph forms no cuspids or self intersections), allowing us to define a group law as the one described in the next paragraph. Henceforth, we will only consider elliptic curves reduced to the short Weierstrass form.

3.1.2 Group Law and Cardinality

In this section we will consider elliptic curves in the affine Weierstrass equation. The same addition law is also valid for curves in the projective Weierstrass equation.

We denote the set of points of an elliptic curve E defined over a field K as $E(K)$. Two points $P = (x, y), P' = (x', y') \in E(K)$ are equivalent if there exists a $c \in K^\times$ such that $x = cx', y = cy'$. The set of points $E(K)$ is defined as:

$$E(K) = \{(x, y) \in K^2 : y^2 = x^3 + ax + b\} \cup \{O\}$$

It can be shown (e.g., [8]-Section III.3) that the set $E(K)$ forms an additive abelian group. For all $P = (x_P, y_P) \in E(K)$ the addition law is defined as follows: first of all, O is the neutral element, i.e., $P + O = O + P = P$. Next, the only inverse of P is $-P = (x_P, -y_P)$, i.e., $P + (-P) = O$. Finally, let $Q = (x_Q, y_Q) \in E(K) \setminus \{O, -P\}$ and let $\lambda \in K$ be determined by $\lambda = (y_P - y_P)/(x_Q - x_Q)$ if $P \neq Q$, and $\lambda = (3x_P^2 + a)/2y_P$ if $P = Q$. Then, $P + Q = R$, where $R = (x_R, y_R)$, with $x_R = \lambda^2 - x_P - x_Q$ and $y_R = \lambda(x_P - x_R) - y_P$. The above group law can be interpreted geometrically and follows the chord-and-tangent principle. Namely, a projective line passing through two points of the curve always intersects the curve at a 3rd point, and the reflection of this point on the x-axis gives the sum of the earlier two points (Figure 3.1[1]).

[1] source:https://www.semanticscholar.org/paper/Applying-Pell-Numbers-for-Efficient-Elliptic-Curve-Du
d2cbadf7552ea7dff2612ce7c44fb95920109e23

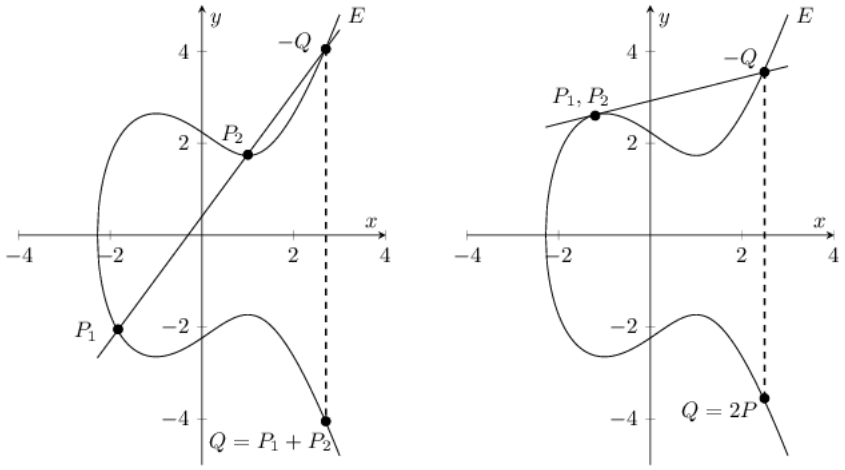

Figure 3.1: Addition law: on the left we compute $P_1 + P_2$ with $P_1 \neq P_2$, on the right we compute $P_1 + P_2$ with $P_1 = P_2$.

For $n \in \mathbb{Z}$ and $P \in E$, we denote as $[n]P$ the point obtained by adding n copies of P (when $n > 0$) or $-n$ copies of $-P$ (when $n < 0$). Namely:

$$[n]P = \begin{cases} \overbrace{(P + P + ... + P)}^{n \text{ times}} & \text{if } n > 0 \\ O & \text{if } n = 0 \\ \overbrace{(-P - P - ... - P)}^{-n \text{ times}} & \text{if } n < 0 \end{cases}$$

When an elliptic curve E is defined over a finite field $\mathbb{F}_q$ with $char(\mathbb{F}_q) = p$, the set of $\mathbb{F}_q$ rational points $E(\mathbb{F}_q)$ of E is a finite group of order:

$$\#E(\mathbb{F}_q) = q + 1 - t \tag{3.2}$$

where $|t| \leq 2\sqrt{q}$ is the trace of Frobenius of E. When $t \equiv 0 \mod p$, we say that E is *supersingular*, otherwise we say that E is *ordinary*. When E is defined over $\mathbb{F}_p$ with p prime and $t \equiv 1 \pmod p$, we say that E is *anomalous*.

3.1.3 Isomorphisms and Endomorphisms rings

Isomorphic classes of elliptic curves are labeled using *j-invariants*. We define j-invariants as follows:

Definition 3.1.1 (j-invariant) *Given an elliptic curve $E : y^2 = x^3 + ax + b$ defined over a field K, the j-invariant of E is the field element:*

$$j(E) = \frac{1728 \cdot 4a^3}{4a^3 + 27b^2}.$$

Two elliptic curves are isomorphic if and only if they share the same j-invariant. Such isomorphism is defined over $\bar{K}$, the algebraic closure of K. Given a j-invariant $j \in K \setminus \{0, 1728\}$, one can define an elliptic curve having j as j-invariant as follows:

$$E(j) : y^2 = x^3 + a(j)x - a(j) \qquad \text{with } a(j) = \frac{27j}{4(1728 - j)}. \tag{3.3}$$

Notice that the point $P = (1, 1) \in E(j)$ for any $j \in K \setminus \{0, 1728\}$.

When $j \in \{0, 1728\}$, we can first sample a point $P = (u, v)$ with $u \neq 0$, $u^2 \neq v^3$, and $u, v \in K$. Then we set:

$$E(0) : y^2 = x^3 + v^2 - u^3, \qquad E(1728) : y^2 = x^3 + \frac{v^2 - u^3}{u}x.$$

For any elliptic curve E, we denote the n-torsion subgroup $E[n]$ to be the set of points on an elliptic curve of order dividing n:

$$E[n] := \{P \in E : nP = O\} \tag{3.4}$$

An *isogeny* from two elliptic curves E to E' is a rational map $\phi : E \to E'$ which is also a group morphism. An *endomorphism* of an elliptic curve E is an isogeny from E to itself. The *endomorphism ring* of E, denoted $End(E)$, is the set of all endomorphisms of E, with addition being defined pointwise and multiplication defined by composition. We give three examples of endomorphisms, taken from [9] and [10]:

Example 3.1.1 (multiply-by-n) *The multiply-by-n map, which sends a point $P \in E$ to $nP \in E$ is an endomorphism. In particular, every endomorphism ring of an elliptic curve contains a subring isomorphic to $\mathbb{Z}$, since we can apply a multiply-by-n map for any integer $n \in \mathbb{Z}$.*

Example 3.1.2 (Frobenius) *Let E be an elliptic curve defined over $\bar{\mathbb{F}}_q$. The q-th power map $\phi : (x, y) \in E(\bar{\mathbb{F}}_q) \to (x^q, y^q) \in E(\bar{\mathbb{F}}_q)$ is an endomorphism and takes the name of Frobenius endomorphism. Furthermore, if $P \in E(\mathbb{F}_q)$ then $\phi(P) = P$, i.e., when $P \in E(\mathbb{F}_q)$ the Frobenius endomorphism corresponds to the identity map (the inverse is also true).*

Example 3.1.3 (special endomorphism) *The elliptic curve $E : y^2 = x^3 + x$ has an endomorphism $\phi(x, y) = (-x, iy)$, with i being the imaginary unit. Since $(iy)^2 = (-x)^3 + (-x)$, ϕ^4 is the identity map, but (x, y) is not a point of any particular order, so ϕ is not a multiply-by-n map.*

While the first two endomorphisms are common to all the elliptic curves defined over $\mathbb{F}_q$, only a few curves have an endomorphism like the last one, i.e., all the curves having 1728 as j-invariant. As we will see in Section 3.2, elliptic curves having non-trivial endomorphisms can be used to build curves with arbitrary orders.

3.1.4 Quadratic fields and orders

Here, basing on the definitions provided by [6, 11], we give the definitions of quadratic field, order of a quadratic field and the relation between their discriminants.

A field K is said to be a *quadratic field* if it has degree two over $\mathbb{Q}$. A quadratic field $\mathbb{Q}(\sqrt{d})$ is obtained by adding the square root of (the discriminant) d to the field of rational numbers. Every element $e \in \mathbb{Q}(\sqrt{d})$ can be written as $e = a + b\sqrt{d}$, for $a, b \in \mathbb{Q}$. The discriminants of quadratic fields are said to be *fundamental*: an integer $d \in \mathbb{Z}$ is a fundamental discriminant if it satisfies either $d \equiv 1 \pmod 4$ and d is square-free or $d = 4D$, with D square-free and $D \equiv 2, 3 \pmod 4$. We will denote a quadratic field K of fundamental discriminant d as $K = \mathbb{Q}(\sqrt{d})$. If $d < 0$, we call it an *imaginary quadratic field*.

Given a number field K, we define its ring of integers $\mathcal{O}_K$ to be the the integral closure of $\mathbb{Z}$ in K. In particular, $\mathcal{O}_K$ consists of all the elements of K satisfying monic polynomials in $\mathbb{Z}[x]$. Now, let $\mathcal{O}_K$ denote the ring of integers of a quadratic field K. A subring $\mathcal{O} \subset \mathcal{O}_K$ is said to be an *order* if it is a free $\mathbb{Z}$ module of rank 2 containing an integral base of K. Every order of a quadratic field is associated with a non-square discriminant $D \in \mathbb{Z}$ of the form $D \equiv 0, 1 \pmod 4$.

There is a relation between fundamental discriminant in quadratic fields and discriminant of orders. Notably, every order $\mathcal{O}$ of the quadratic field $\mathbb{Q}(\sqrt{d})$ has discriminant $D = df^2$, where $f \in \mathbb{N}$ is the conductor of $\mathcal{O}$. Conversely, every discriminant D can be written uniquely as $D = df^2$, and there exists a unique order (up to isomorphism) $\mathcal{O}$ of $\mathbb{Q}(\sqrt{d})$ of discriminant D.

3.2 Complex multiplication

In this section, we will study elliptic curves with CM and discuss how CM theory can be used to construct elliptic curves that have a specified group order when reduced over a finite field. Firstly, we define CM curves (Section 3.2.1), and we describe the relationship between j-invariants and the Hilbert class polynomial (Section 3.2.2). Then, we describe how to generate supersingular and ordinary curves (Section 3.2.3). We end this section by describing how to generate anomalous curves (Section 3.2.4).

3.2.1 Curves with Complex Multiplication

The endomorphism ring of an elliptic curve E defined over a field K with characteristic 0, is either isomorphic to $\mathbb{Z}$ or to an order in an imaginary quadratic field. In the latter case, we say that E has *complex multiplication*

by that order. Notice that when E is defined over a finite field, its endomorphism ring is always larger than $\mathbb{Z}$. In particular, it corresponds either to an order in a quadratic field (for *ordinary* curves), or to a maximal order in a quaternion algebra (for *supersingular* curves).

3.2.2 Hilbert Class Polynomial and j-invariants

Let $K = \mathbb{Q}(\sqrt{-d})$ be an imaginary quadratic field and $\mathcal{O}$ be an order of K of discriminant $-D$. There is an important relationship between elliptic curves with CM and the Hilbert class polynomial $H_{-D}(x)$. In particular, the roots of $H_{-D}(x)$ are exactly the j-invariants of elliptic curves having CM by $\mathcal{O}$. More formally ([12]-Lecture 21):

Theorem 3.2.1 *Let $\mathcal{O}$ be an order in some imaginary quadratic field $\mathbb{Q}\sqrt{-d}$, where $-d < 0$ is a fundamental discriminant, and $-D$ be the discriminant of $\mathcal{O}$. Let E be an elliptic curve defined over the number field $\mathbb{L}$ that has complex multiplication by $\mathcal{O}$. Then, from class field theory, we know that all the j-invariants of elliptic curves having complex multiplication by $\mathcal{O}$ are exactly the roots of a polynomial $H_{-D}(x) \in \mathbb{Z}[x]$, called Hilbert class polynomial, which completely splits in $\mathbb{L}$. Moreover, $H_{-D}(x)$ is irreducible over $\mathbb{Q}$.*

Hence, after computing the roots of H_{-D}, we can use them as j-invariants in (3.3) to construct elliptic curves that have complex multiplication by an order of discriminant $-D$. In the next paragraph, this method will turn out to be very useful.

3.2.3 Generating Supersingular Curves

Before describing how to generate supersingular curves, we need to introduce the following definition.

Definition 3.2.1 *Let $\mathcal{O}_K$ be the ring of integers of the imaginary quadratic field K, defined as above. The ideal $(p) = p\mathcal{O}_K$ generated by an odd prime number $p \in \mathbb{Z}$ can behave in three different ways:*

1. *We say that p splits when there exists distinct primes $\mathfrak{p}, \mathfrak{q} \in \mathcal{O}_K$ such that $(p) = \mathfrak{pq}$ (i.e., $(p) = p\mathcal{O}_K$ is a product of two distinct prime ideals).*
2. *We say that $\mathfrak{p}$ is ramified when there exists a prime $\mathfrak{p} \in \mathcal{O}_K$ s.t. $(p) = u\mathfrak{p}^2$, where u is a unit of K (i.e., (p) is the square of a prime ideal)*
3. *We say that p is inert when there exists a prime $\mathfrak{p} \in \mathcal{O}_K$ such that $(p) = \mathfrak{p}$ (i.e., (p) is a prime ideal).*

Each behavior of p happens on specific values of the Kronecker symbol evaluated between p and the discriminant $D = df^2$ (e.g., [11]-Page 57). Namely:

1. p *splits* $\iff (D/p) = 1$ (i.e., D is a quadratic residue modulo p)
2. p is *ramified* $\iff (D/p) = 0$ (i.e., p divides D)
3. p is *inert* $\iff (D/p) = -1$ (D is not a quadratic residue modulo p)

Let now E be an elliptic curve defined over a number field $\mathbb{L}$ having CM by an order of K. When E has good reduction for a given prime p (i.e., E is non-singular when reduced modulo p), the behavior of p in K determines whether $\bar{E} = E/\mathbb{F}_p$ is *ordinary* or *supersingular*. More precisely, we have the following theorem ([12]-Lecture 22):

Theorem 3.2.2 *: Let E be an elliptic curve defined over a number field $\mathbb{L}$ whose endomorphism ring is an order $\mathcal{O}$ in an imaginary quadratic field $\mathbb{Q}(\sqrt{-d})$, and p be a prime number not dividing the conductor of $\mathcal{O}$. Let $\mathfrak{p}$ be a prime of $\mathbb{L}$ above p where E has good reduction. Then $E \mod \mathfrak{p}$ is ordinary if and only if p splits in $\mathbb{Q}(\sqrt{-d})$. Moreover, when this is the case, then there exists u such that $t^2 - 4q = -u^2 d$, where $t = q + 1 - \#E(\mathbb{F}_q)$, $f|u$ and $\mathbb{F}_q$ is the base field of j.*

Hence, for a root j of $H_{-D}(x)$ we have that $E(j)$ is ordinary if $(-D/p) = 1$ and supersingular if $(-D/p) = 0, -1$ (assuming that $E(j)$ has a good reduction at p). Actually, as we consider p of cryptographic size and the computation of the Hilbert polynomial is extremely expensive for large discriminants, we will never be in the case $(-D/p) = 0$.

We now give concrete examples on how to build ordinary and supersingular curves using CM theory.

Example 3.2.1 (Supersingular curve) *Our goal is to construct a supersingular elliptic curve E over $\mathbb{F}_p$ (i.e., $\#E(\mathbb{F}_p) = p + 1$), for some prime p. We start by fixing a discriminant (say $D = 11$) and picking some primes $p \in \mathbb{Z}$ of the desired dimension until $(-D/p) = -1$. According to Theorem 3.2.2, if E has a good reduction for such p then we will be able to construct a supersingular curve over $\mathbb{F}_p$. The Hilbert class polynomial $H_{-D}(x) = x + 32768$ has degree one, and $j_{-D} = -32768$ is the only root. Now, we compute $E(j_{-D})$ as in (3.3), and obtain:*

$$E : y^2 = x^3 + a(j_{-D})x - a(j_{-D})$$

Lets now consider $\bar{E} = E/\mathbb{F}_p$ with equation:

$$\bar{E} : y^2 = x^3 + ax + b \pmod{p}$$

Take $P = (1, 1) \in \bar{E}(\mathbb{F}_p)$, one can check[2] that $[p + 1]P = O$, and $\#\bar{E}(\mathbb{F}_p) = p + 1$.

[2] e.g., with `https://doc.sagemath.org/html/en/reference/arithmetic_curves/sage/schemes/elliptic_curves/ell_finite_field.html`

Parameters:
$a(j_{-D}) = -3456/539$
$p = 569092308043786224773165726542006491001025686599699024807530493704302798403300305804259479798462825391360393453616584379727356736423516301930219347990660 9$
$a = 168932781608545075999455503240669830352809109194715851519860628928921053329365582427980550589525142973316628854505850650753946341053362909663887004969392,$
$b = 552199029882931717173220176217939507965744775680227439655544430811410693070363747561461424739510311094028730568165999314651962102318180010963830647493721 7.$

Example 3.2.2 (Ordinary curve) *As before, we start by selecting a discriminant (say $D = 19$) and a prime p. Our goal is to construct an elliptic curve E s.t., $\#E(\mathbb{F}_p) \neq p + 1$. In this case, to be able to build an ordinary curve we need $(-D/p) = 1$. The Hilbert class polynomial $H_{-D}(x) = x + 884736$ has degree one, and its only root is $j_D = -884736$. Now, compute $E(j_D)$ as in (3.3) and obtain:*

$$E : y^2 = x^3 + a(j)x - a(j)$$

Lets now consider $\bar{E} = E/\mathbb{F}_p$. Its equation is:

$$\bar{E} : y^2 = x^3 + ax + b \pmod{p}$$

We see that $P = (1, 1) \in \bar{E}$, and it is possible to check that $[p + 1]P \neq O$, hence $\bar{E}$ is ordinary.
Parameters:
$a(j) = -128/19$
$p = 917559200205229353201584877192279345041825259711899293553791149789881125318228646791804846555446114855553650251291779193136438721328725196018673148600074 1$
$a = 386340715875886043453298895659907092649189583036589176233175220964160473818201535491286251181240469412864694842649170186583763672138410608849967641515820 0$
$b = 531218484329343309748285981532372252392635676675310117320615928825720651500027111300518595374205645442688955408642609006552675049190314587168705507084254 1.$

3.2.4 Constructing Anomalous Curves

In the paragraph above we discussed how to generate supersingular and ordinary curves using CM theory. Using CM theory it is also possible to

generate ordinary curves with a specific order. Here we describe, as shown in [13], how to construct anomalous elliptic curves over $\mathbb{F}_p$, where p is a specially selected prime number.

Let E be an elliptic curve defined over $\mathbb{F}_p$, then the number of its rational point is given by (3.2). We define the *quadratic twist* $\tilde{E}$ of the elliptic curve $E:\ y^2 = x^3 + ax + b$ as follows:

$$\tilde{E}:\ y^2 = x^3 + u^2 ax + u^3 b$$

for any non-quadratic residue $u \in \mathbb{F}_p$. From Definition 3.1.1, we easily verify that E and $\tilde{E}$ share the same j-invariant. Furthermore, the number of their rational points is related by the equation:

$$\#E(\mathbb{F}_p) + \#\tilde{E}(\mathbb{F}_p) = 2p + 2. \tag{3.5}$$

Moreover, if the prime number p defining the field satisfies $4p = x^2 + Dy^2$ for some $x, y \in \mathbb{Z}$, and if E has complex multiplication by an order of discriminant $-D$, the trace of the Frobenius endomorphism $t = \pm x$. Thus, the orders of E and its quadratic twist over $\mathbb{F}_p$ will be $p+1-x$ and $p+1+x$ respectively (since their sum is $2p+2$). In particular, if we choose $x = 1$ we either have $\#E(\mathbb{F}_p) = p$ and $\#\tilde{E}(\mathbb{F}_p) = p + 2$ or the opposite (i.e., either E or $\tilde{E}$ is anomalous). Consequently, when p satisfies the following equation:

$$4p = 1 + Dy^2 \tag{3.6}$$

for some $y \in \mathbb{Z}$, we can get the j-invariant of elliptic curves with complex multiplication by an order of discriminant $-D$ by finding the roots of the Hilbert polynomial $H_{-D}(x)$, and generate one of such curves E using (3.3).

Last, we have to check whether E or its quadratic twist $\tilde{E}$ is anomalous. To this end, we take a point $P \in E$ and check whether $[p]P = O$. If the latter holds, then E is anomalous, otherwise its quadratic twist $\tilde{E}$ is anomalous.

Let $p > 2$ be a prime number, then not all the discriminants $-D$ are compatible with (3.6). First of all, as $-4p+1$ is odd, we must have both $-D$ and y odd, meaning that we should only consider discriminants of the form $-D \equiv 1 \pmod 4$. Furthermore, if we consider the equation $-4p = -1 - Dy^2 \pmod 8$, we notice that we have a solution only when $D \equiv 3 \pmod 8$. Finally, we can rewrite $y = (2m+1)$ and have $4p = 1 + D(2m+1)^2$, which implies that our prime should be of the form:

$$p = Dm(m+1) + \frac{D+1}{4}, \quad \text{with } D \equiv 3 \pmod 8 \tag{3.7}$$

We give now an example of the above procedure.

Example 3.2.3 (Anomalous curve) *In this example, our goal is to construct an anomalous elliptic curve over the finite field $\mathbb{F}_p$ (with p prime).*

Lets start with the discriminant $D = 163$. *For our curve to be anomalous, we need our prime to be of the form* $p = 163m(m+1) + 41$, *with* $m > 0$. *Hence, we start generating random numbers* $m \in \mathbb{Z}$ *of around half the desired dimension, until* p *is a prime number. After obtaining such* p *(see below), we compute the Hilbert class polynomial* $H_{-D}(x) = x + 262537412640768000$ *and notice it has degree one, with a single root* $j_D = -262537412640768000$. *Now, we compute* $E(j_D)$ *as in* (3.3), *and obtain:*

$$E : y^2 = x^3 + a(j)x - a(j)$$

Lets now consider $\bar{E} = E/\mathbb{F}_p$, *its equation is:*

$$\bar{E} : y^2 = x^3 + ax + b \pmod{p}$$

We know that either $\bar{E}$ *or its quadratic twist* $\tilde{E}$ *have cardinality* p. *Take* $P = (1, 1) \in \bar{E}$, *we have that* $[p]P = O$. *Hence,* $\bar{E}$ *is anomalous, while* $\tilde{E}$ *has cardinallity* $p + 2$.
Parameters:
$a(j) = -37982843264000/5627087890963$
$p = 708325724819030482802657584153088004633593399843668232573225116965780839269531032171550091078419379399929563861978554399785007796821658136989217260608407807$
$a = 111828962985585971874083199736073690575707856107867104477843115765310501552925367930476522374658058659444494823038621039853687054364116608918236403671014573$
$b = 596496761833444510928574384417014314057885543735801128095382001200470337716605664241073568703761320740485069038939933359931320742457541528070980856937393234.$

Chapter 4

Elliptic Curve Factorization Methods

In this Chapter, we first present the ECM algorithm (Section 4.1). Then, we present a $p+1$ version of ECM with supersingular curves (Section 4.2), and a version of ECM with anomalous curves (Section 4.3). We end the chapter with some considerations on the implementation (Section 4.4).

4.1 ECM

In this section, we will describe the ECM algorithm proposed by H.W.Lenstra in 1987 [1].

Given an RSA modulus $n = pq$, the goal of ECM is solving the factorization of n. Essentially, ECM is based on the same idea as Pollard's $p-1$ algorithm that we discussed in Section 2.1. What differentiates the two approaches is that, instead of using the multiplicative group $\mathbb{Z}_p^\times$, ECM works in the additive group of a randomly generated elliptic curve E defined over the finite field $\mathbb{F}_p$. Consequently, as the cardinality of $\mathbb{Z}_p^\times$ is fixed at $p-1$, whenever $p-1$ is not B-powersmooth Pollard's method fails. On the contrary, the additive order of a random elliptic curve $E/\mathbb{F}_p$ is distributed between $p+1\pm 2\sqrt{p}$, as stated by Hasse's theorem (3.2). Instead of failing, as soon as we realize that both $\#E(\mathbb{F}_p)$ and $\#(\mathbb{F}_q)$ are not B-powersmooth, we can retry with a different curve, and hope that the new curve will now have smooth order.

More precisely, the algorithm proceeds as follows: first, we pick a random elliptic curve $E : y^2 = x^3 + ax + b$ with coefficients over $\mathbb{Z}_n$ and a point $P = (x_P, y_P)$ in it. As extracting square roots in $\mathbb{Z}_n$ is a hard problem, we first pick the a coordinate of the curve, and the x_P, y_P coordinates of the point. Then, we set $b = y_P^2 - x_P^3 - ax_P$ so that $P \in E$. Next, we compute $m = \prod_{i=1}^{k} p_i^{\lfloor \log_{p_i}(B) \rfloor}$ for all primes $p_i \leq B$, and we split m in the list $M_s = [m_1, m_2, ..., m_j]$, as we did in (2.3). Finally, we compute $[m]P$ by

iteratively running the by-products $[m_i]P$ for each $m_i \in M_s$. If an inversion error occurred, then we just factored n[1], otherwise we pick another curve and repeat.

Algorithm 1: ECM

input : An RSA Modulus $n = pq$, a bound B, and a time limit w
output: The factorization of n or **failed**

Compute $m = \prod_{i=1}^{k} p_i^{\lfloor \log_{p_i}(B) \rfloor}$ and split it into $M_s = [m_1, ..., m_j]$
while *elapsed time* $< w$ **do**
  Pick a, x_p, y_p randomly from $\mathbb{Z}_n$
  Set $b = y_p^2 - x_p^3 - ax_p$
  Set $E : y^2 = x^3 + ax - b \pmod{n}$, and $P = (x_p, y_p) \in E$
  foreach $m_i \in M_s$ **do**
    try
      $P = [m_i]P$
    if *a divisor D is not invertible in* $\mathbb{Z}_n$ **then**
      return $gcd(D, n)$
    end
  end
end
return failed

The algorithm has asymptotic complexity $L_p(1/2, \sqrt{2})$, which is sub-exponential in p. Despite ECM is not the most efficient algorithm known (e.g., Number Field Sieve has complexity around $L_n(1/3, 2)$), its complexity is dependent on the smallest factor of n. In fact, due to Hasse's bound, the more n is unbalanced the faster ECM will succeed, as $\#E(\mathbb{F}_p) \simeq p$ has more chances of being smooth.

4.2 ECM with Supersingular Curves

In this section, we describe a variant of ECM based on supersingular elliptic curves that work as a $p+1$ factorization algorithm. First, we recall the concept of supersingular elliptic curves, and next we move on to the description of the algorithm.

4.2.1 Supersingular Elliptic Curves

We recall that a supersingular elliptic curve is a curve E defined over a finite field $\mathbb{F}_q$ of prime characteristic p, such that the trace of the Frobenius

[1] unless both $\#E(\mathbb{F}_p), \#E(\mathbb{F}_q)$ are B-powersmooth, in such case we should increase the bound B

endomorphism $t \equiv 0 \pmod{p}$. From (3.2), we see that, when E is defined over $\mathbb{F}_p$, the number of its rational points is $\#E(\mathbb{F}_p) = p + 1$.

4.2.2 The ECM p+1 algorithm

Let $n = pq$ be an RSA modulus such that $p + 1$ is B-powersmooth. In this section, we show that by using a list of potential supersingular curves, we can create a $p + 1$ version of ECM which factors n with high probability.

First of all we need to generate supersingular curves over $\mathbb{F}_p$. In Section 3.2.3, we saw that, when p is *inert* in $\mathbb{Q}(\sqrt{-D})$, solving the Hilbert polynomial $H_{-D}(x) \pmod{p}$ produces the j-invariants of supersingular curves in $\mathbb{F}_p$. Our first impediment is that we do not know p beforehand, hence, we can only attempt to find solutions modulus n. In general, finding the roots of $H_{-D}(x) \pmod{n}$ without knowing the factorization of n is a hard problem and thereby very inefficient. However, by carefully choosing the discriminant $-D$ (see Table 4.1), we can restrict ourself to the case where the Hilbert polynomial has degree one, and its single root is independent from n. Once we have a candidate j-invariant, we use the procedure described in Section 3.2.3 to generate a supersingular elliptic curve E such that the point $P = (1, 1) \in E$.

Note that, if P was a m-torsion point over $\mathbb{Q}$ on E, reducing it modulus n would still return an m-torsion point. Hence, $P \pmod{n}$ would lay in a subgroup and would not be useful for factoring n, as plugging it into the ECM algorithm would not cause an inversion error but return the actual point at infinity. However, a result from Mazur ([14]-Theorem 4.1) implies that on an elliptic curve defined over $\mathbb{Q}$, the order of any torsion point is in $T = \{1, 2, 3, ..., 10\} \cup \{12\}$. By evaluating the n-th division polynomial $\psi_n(a(j))$ of the curve $E(j) : y^2 = x^3 + a(j)x - a(j)$ at $P = (1, 1)$, one can show that for every $n \in T$ we have $[n]P \neq O$ (as $\psi_n(a(j))$ has no solutions in $\mathbb{Q}$). Hence, for all the curves $E(j) : y^2 = x^3 + a(j)x - a(j)$ defined over $\mathbb{Q}$, the point $P = (1, 1)$ is a non torsion point.

Let $E(j)/\mathbb{Q}$ be an elliptic curve of non-zero rank having as j-invariant the solution of one of the Hilbert polynomials $H_{-D}(x)$ contained in Table 4.1 such that $(-D/p) = -1$. Moreover, let $P \in E(j)$ be a non torsion point of the curve, $\bar{E} = E \pmod{n}$ and $\bar{P} = P \pmod{n}$. Then, assuming that $p+1$ is B-powersmooth and $q + 1$ is not, we have:

$$\begin{aligned} [m]\bar{P} &= O \pmod{p} \\ [m]\bar{P} &\neq O \pmod{q} \end{aligned}$$

with $m = \prod_{i=1}^{k} p_i^{\lfloor \log_{p_i}(B) \rfloor}$ as in Section 2.2. This implies that running the ECM algorithm on the curve $\bar{E}$ and the point $\bar{P}$ yields a non-trivial factor of n. We highlight that, as the j-invariant is independent of n, the same curve

$-D = -d \cdot f^2$	$H_{-D}(x)$	j
$-3 = -3 \cdot 1$	x	0
$-12 = -3 \cdot 4$	$x - 54000$	54000
$-27 = -3 \cdot 9$	$x + 12288000$	-12288000
$-4 = -4 \cdot 1$	$x - 1728$	1728
$-16 = -4 \cdot 4$	$x - 287496$	287496
$-7 = -7 \cdot 1$	$x + 3375$	-3375
$-28 = -7 \cdot 4$	$x - 16581375$	16581375
$-8 = -8 \cdot 1$	$x - 8000$	8000
$-11 = -11 \cdot 1$	$x + 32768$	-32768
$-19 = -19 \cdot 1$	$x + 884736$	-884736
$-43 = -43 \cdot 1$	$x + 884736000$	-884736000
$-67 = -67 \cdot 1$	$x + 147197952000$	-147197952000
$-163 = -163 \cdot 1$	$x + 262537412640768000$	-262537412640768000

Table 4.1: Hilbert Polynomials of degree one and associated j-invariants

can also be used to factor other RSA moduli that have a factor p' such that $p' + 1$ is B-powersmooth and p' is inert in $\mathbb{Q}(-D)$.

As we have no way to test whether $(-D/p) = -1$ without knowing the factorization of n, we would need to repeat the above procedure for all j-invariants in Table 4.1, and hope that at least one of them can be used to generate a supersingular curve. Actually, as the discriminant $-D = -df^2$ is given by the product between the fundamental discriminant and a square, the condition $(-D/p) = -1$ can be reduced to $(-d/p) = -1$. Hence, we can reduce the number of candidate curves by taking just one j-invariant for each fundamental discriminant $-d$. In the end, we can just consider the following list of j-invariants: $J_s = [54000, 287496, -3375, 8000, -32768, -884736, -884736000, -147197952000, -262537412640768000]$.

The above algorithm has complexity $O(B)$ and succeeds when $p+1$ is B-powersmooth and p is inert in $\mathbb{Q}\sqrt{-d}$, for some $d \in \{3, 4, 7, 8, 11, 19, 43, 67, 163\}$. If we assume that each fundamental discriminant $-d$ has roughly $1/2$ probability of being a quadratic residue modulo p, Algorithm 2 succeeds with probability $p_{succ} = 1 - (0.5)^9 \simeq 0.998$.

4.3 ECM with Anomalous Curves

In this section, we present an efficient variant of ECM with anomalous curves that efficiently factors some specific type of RSA moduli (Section 4.3). We remark that similar algorithms have already been proposed by [5, 6].

Algorithm 2: ECM with Supersingular Curves

input : An RSA modulus $n = pq$ and a bound B
output: The factorization of n or **failed**

Set $J_s = [54000, 287496, -3375, 8000, -32768, -884736, -884736000, -147197952000, -262537412640768000]$
foreach $j \in J_s$ **do**
  Set $a(j) = \frac{27j}{4(1728-j)}$
  Set $E : y^2 = x^3 + a(j)x - a(j)$ and $P = (1, 1) \in E$
  Set $\bar{E} : E \pmod n$ and $\bar{P} = P \pmod n$
  if `ECM` *executed on* $\bar{E}$,$\bar{P}$, *and* B *succeeds* **then**
    return p,q
  end
end
return failed

4.3.1 The ECM Anomalous Algorithm

We recall that *anomalous* curves are elliptic curves having *exactly* p rational points over $\mathbb{F}_p$ (Section 3.1.2). In Section 3.2.3, we showed that given a discriminant $-D \equiv 3 \pmod 8$ associated with some order in a quadratic field $\mathbb{Q}(\sqrt{-d})$ and such that $p = m(m+1)\frac{D+1}{4}$ for some integer $m > 0$, we have a method based on CM to generate anomalous elliptic curves over $\mathbb{F}_p$. However, we are only given an RSA modulus $n = pq$ and—as in the supersingular case—we need to solve the Hilbert polynomial $H_{-D}(x) \pmod n$ without knowing the factorization of n.

Again, we could just select discriminant for which $H_{-D}(x)$ has degree one, with the additional condition $-D \equiv 3 \pmod 8$. With this approach, we will be able to build anomalous curves for primes of the form $p = Dm(m+1) + \frac{D+1}{4}$ only when $-D \in \{-3, -11, -19, -27, -43, -67, -163\}$ (see Table 4.2). Alternatively, we can avoid finding an explicit solution of $H_{-D}(x) \pmod n$ and work with a symbolic solution j instead. The advantage here is that we have no limitation on the value of the discriminant $-D$.

Say that we take the latter strategy, our curves will now be defined over the number field $\mathbb{Q}(j) \coloneqq \mathbb{Q}[x]/H_{-D}(x)$ that we can naturally reduce to the ring $(\mathbb{Z}/n\mathbb{Z})[x]/H_{-D}(x)$ when executing the ECM algorithm. For this to work, we need at least one curve to be defined over $\mathbb{F}_p$, i.e., we need $H_{-D}(x)$ to have one root $j_{-D} \in \mathbb{F}_p$. It turns out (e.g., [15]-Section 5.1) that if p splits in $\mathbb{Q}(d)$ and is such that $4p = t^2 - dy^2$ then $H_D(x)$ splits completely over $\mathbb{F}_p$, meaning that $H_D(x)$ always has a valid solution $j \in \mathbb{F}_p$. Note that in our case $d < 0$, $t = 1$, and $y = (2m+1)f$. Another issue is that after replacing the symbolic j with a root $j_{-D} \in \mathbb{F}_p$, we have no guarantees on whether $E(j_{-D})/\mathbb{F}_p$ is an anomalous curve or its quadratic twist (as they both share

$-D$	j	Anomalous p
-3	0	$3m(m+1)+1$
-11	-32768	$11m(m+1)+3$
-19	-884736	$19m(m+1)+5$
-27	-12288000	$27m(m+1)+7$
-43	-884736000	$43m(m+1)+11$
-67	-147197952000	$67m(m+1)+17$
-163	-262537412640768000	$163m(m+1)+41$

Table 4.2: Discriminants of anomalous curves and associated primes

the same j-invariant). In particular, any integer $x \in \mathbb{Z}_p$ is either the x-coordinate of a point on the elliptic curve $E(j_D)$ or on its quadratic twist $\tilde{E}(j_{-D})$. As both cases happen with half probability, one can iteratively pick random x-coordinates from $\mathbb{Z}_n$ until being reasonably confident that at least one of them is the x-coordinate on an anomalous curve (when reduced modulus p).

Let $n = pq$ be an RSA modulus such that p has the form $p = Dm(m+1) + \frac{D+1}{4}$ for some discriminant $-D$. Let j be a symbolic root of the Hilbert polynomial $H_{-D}(x)$, let E be an elliptic curve of non-zero rank defined over $\mathbb{Q}(j)$ and $P \in E$ a point on it. If E is anomalous when reduced over $\mathbb{F}_p$ (and not over $\mathbb{F}_q$), then we have:

$$[n]\bar{P} = O \pmod{p}$$
$$[n]\bar{P} \neq O \pmod{q}$$

with $\bar{E} = E \pmod{n}$, and $\bar{P} = P \pmod{n}$. Hence, $[n]\bar{P}$ computed with the ECM method yields the factorization of n. As a final remark, we underline that, as we just know the x-coordinate of the point P, we need to perform all the arithmetic operations on the curve using the XZ-notation [2] $P = (x : 1)$ instead of the usual XY-notation $P = (x, y)$.

4.4 Implementation

At the beginning of the project, we developed a proof-of-concept implementation of both the ECM $p+1$ algorithm and Williams $p+1$ algorithm using SageMath [3]. We compared the results of the two implementations on an Apple M1 chip for 100 different moduli $n = pq$ of 1024 bits each and generated such that $p+1$ was B-powersmooth for a bound $B = 2^{20}$. Williams algorithm (executed with a limit of 10 repetitions in case of failures) managed to factor all the 100 moduli in an average time of 4.8 seconds each.

[2] e.g., see `https://hyperelliptic.org/EFD/g1p/auto-shortw-xz.html`

[3] `https://www.sagemath.org`

Algorithm 3: Anomalous ECM

input : An RSA Modulus $n = pq$, a list H_s of Hilbert polynomials and an iteration bound K
output: The factorization of n or **failed**

foreach $H_{-D}(x) \in H_s$ **do**
  Set $R[j] = (\mathbb{Z}/n\mathbb{Z})[x]/H_{-D}(x)$
  Set $a(j) = \frac{27j}{4(1728-j)} \in R[j]$
  Set $\bar{E} : y^2 = x^3 + a(j)x - a(j)$
  for $i = 1$ **to** K **do**
    Pick $x_P \in \mathbb{Z}_n$ randomly
    Set $\bar{P} = (x_P : 1) \in \bar{E}$
    try
      $\bar{P} = [n]\bar{P}$
    if *a divisor D is not invertible in $\mathbb{Z}_n$* **then**
      return $gcd(D, n)$
    end
  end
end
return failed

The ECM supersingular implementation was one order of magnitude slower when executed on a single curve (around 30 seconds), and took up to several minutes to factor some of the given integers (as it had to try with multiple curves). Furthermore, all the RSA moduli such that p did not satisfy $(-D/p) = -1$ for any discriminant in Table 4.1 could not be factored. This result gave us experimental evidence that the Williams $p+1$ method is still to be preferred to our supersingular version of ECM.

In the second part of the project, we developed a proof-of-concept implementation of the anomalous ECM algorithm in SageMath, for both the cases we discussed in Section 4.3 (i.e., when we restrict to the discriminants in Table 4.2 and when we work with symbolic solutions). For more efficiency, we implemented the version with "known" j-invariants (i.e., included in Table 4.2) in C using the GMP multi-precision library [4]. We tested our algorithm with vulnerable 1024 bits RSA modules such that a divisor p was of the form $p = Dm(m+1) + \frac{D+1}{4}$ for $m > 0$ and a discriminant $-D \in \{-3, -11, -19, -27, -43, -67, -163\}$. As expected, the algorithm could factor all the vulnerable modules, and the runtime was in the order of 10^{-2} seconds (for the Sage implementation) and 10^{-3} seconds (for the C implementation). We also tested the ECM anomalous algorithm in the version with symbolic j-invariants. As expected, performing operation modulo the Hilbert polynomial $H_{-D}(x)$ becomes more and more expensive as the

[4] `https://gmplib.org`

degree of $H_{-D}(x)$ increases. To give an idea, the runtime of our SageMath implementation is in the order of 10^{-1} seconds for polynomials of degree 1 (e.g., $-D = -67$) to more than 30 seconds for polynomials of degree 8 (e.g., $-D = -995$).

Chapter 5

Conclusion

In this chapter, we first draw some final conclusion from the project (Section 5.1). Then, we talk about some issues we had to overtake and the knowledge gained from the project (Section 5.2). We end this chapter with a section on the future works (Section 5.3).

5.1 Conclusion

In this project, we explored the topic of integer factorization and performed an in-depth analysis of some important factorization algorithms. Moreover, we used the theory of complex multiplication to generate supersingular elliptic curves and to design a new $p+1$ method based on ECM. Although the algorithm shows a novel application of supersingular curves, we acknowledge that, in practice, more efficient $p+1$ methods are already available (e.g., Williams $p+1$). Finally, using CM theory, we developed a variant of ECM over anomalous elliptic curves that factors a restricted class of RSA moduli in polynomial time. Even though honestly generated RSA moduli would hardly satisfy the condition $p = Dm(m+1) + \frac{D+1}{4}$ for small D (such that the Hilbert polynomial $H_{-D}(x)$ can be efficiently computed and used in the algorithm), this method exposes a vulnerable class of RSA moduli. A malicious entity could corrupt the key generation algorithm [1] to generate those vulnerable RSA moduli and later be able to efficiently retrieve users' secret keys.

5.2 Difficulties and Acquired Knowledge

During this project, we have been confronted with numerous problems, both during the design and the implementation of the algorithms. For example, during the ECM algorithm our curves were defined over the ring $\mathbb{Z}_n$ instead

[1] https://blog.cloudflare.com/how-the-nsa-may-have-put-a-backdoor-in-rsas-cryptography-a-technical-

of the finite field $\mathbb{F}_p$. Hence, to compute the j-invariants of curves with specific order (e.g., anomalous), we had the problem of finding roots to the Hilbert class polynomial modulo n, which is supposed to be a hard problem when n is not prime. As a workaround, we either selected only discriminants $-D$ such that the Hilbert polynomial $H_{-D}(x)$ has degree one (i.e., it has a single root independent from n), or we worked with a symbolic root j and considered curves defined over the number field $\mathbb{Q}(j)$. Once we defined these curves, we still had the problem of finding points on them, which is again a hard problem. When working with a known j-invariant, we just considered the curve $E(j) : y^2 = x^3 + a(j)x - a(j)$ with $a(j) = \frac{27j}{4(1728-j)} \in \mathbb{Q}$, such that the point $P = (1, 1)$ is always available independently of n. When working with a symbolic j-invariant, we switched from the XY notation to the XZ notation, picked random x-coordinates from $\mathbb{Z}_n$ and worked with the point $P = (x : 1)$, knowing that each x-coordinate would either lay on the anomalous curve or its quadratic twist, both with half probability.

One of the most challenging parts of the implementation was writing the C version of the ECM algorithm. As the maximal integer size supported by C is 64 bits, we had to use an external multi-precision library. After some research, we opted for the GMP library, which was by far the most suggested as it is well maintained and documented. It took me some time to gain familiarity with the library, as it has low-level memory management and every arithmetic operation has to be performed with a function (C does not support operators redefinition), which makes the code quite hard to read. We remark that the GMP library is not fully compatible with Apple M1 chips, hence we suggest testing the implementation on other platforms (unfortunately I spent several days trying to debug the code before before realizing that). In the end, we provided a working implementation in the case where the Hilbert polynomial has degree one (i.e., we are not working over number fields). We considered using the Boost C++ library [2] for implementing the case when our curves are defined over a number field, as it provides some support for number theory and polynomials. However, as the supported arithmetic was very limited (e.g., it was not possible to compute the inverse of polynomials or working modulo a polynomial), we had a limited amount of time left and we already had a working implementation in Sage, we decided to leave this step as a possible future improvement.

To conclude, I started this project being already familiar with RSA and the factorization problem but with just a basic understanding of elliptic curves and ECM. At the end of the project, I gained a deeper knowledge of these concepts and familiarity with other mathematical objects (e.g., endomorphism rings, quadratic fields and orders, complex multiplication) that, coming from a CS background, I never encountered before.

[2] `https://www.boost.org`

5.3 Future works

As future works, we are planning to test our implementation of ECM anomalous on special numbers (e.g., Fermat numbers, Mersenne numbers) and test if we can factor some of them with low runtime.

Bibliography

[1] H. W. Lenstra Jr, "Factoring integers with elliptic curves," *Annals of mathematics*, pp. 649–673, 1987.

[2] T. B. Fouotsa, "Factoring integers using supersingular and anomalous curves," in preparation.

[3] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, p. 120–126, feb 1978. [Online]. Available: https://doi.org/10.1145/359340.359342

[4] J. M. Pollard, "Theorems on factorization and primality testing," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 76, no. 3, p. 521–528, 1974.

[5] Q. Cheng, "A new special-purpose factorization algorithm," 2002.

[6] G. Vitto, "Factoring primes to factor moduli: Backdooring and distributed generation of semiprimes," Cryptology ePrint Archive, Paper 2021/1610, 2021, https://eprint.iacr.org/2021/1610. [Online]. Available: https://eprint.iacr.org/2021/1610

[7] H. C. Williams, "A $p + 1$ method of factoring," *Mathematics of Computation*, vol. 39, no. 159, pp. 225–234, 1982. [Online]. Available: http://www.jstor.org/stable/2007633

[8] A. W. Knapp, *Elliptic curves*. Princeton University Press, 1992, vol. 40.

[9] A. Lin, "Complex multiplication and elliptic curves." [Online]. Available: https://web.stanford.edu/~lindrew/18.784p.pdf

[10] J. H. Silverman, *The Arithmetic of Elliptic Curves*, 2nd ed. Springer New York, NY, 2009.

[11] T. Weston, "Algebraic number theory, course notes," 1999.

[12] A. Sutherland, "Lectures - elliptic curves," 2017, https://math.mit.edu/classes/18.783/2017/lectures.html.

[13] F. Leprévost, J. Monnerat, S. Varrette, and S. Vaudenay, "Generating anomalous elliptic curves," *Information Processing Letters*, vol. 93, no. 5, pp. 225–230, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020019004003527

[14] B. Mazur and D. Goldfeld, "Rational isogenies of prime degree," *Inventiones mathematicae*, vol. 44, pp. 129–162, 1978.

[15] Ø. Ø. Thuen, "Constructing elliptic curves over finite fields using complex multiplication," Master's thesis, Institutt for matematiske fag, 2006.