THE ELLIPTIC CURVE FACTORIZATION METHOD

RIBHAV BOSE

Abstract. Due to the modern day importance of cybersecurity and encryption, the question of finding efficient and quick ways of factoring large numbers is increasingly relevant. In this paper we will discuss two different factorization methods, Pollard's $p-1$ algorithm and Lenstra's elliptic curve algorithm. We will start by going over basic algebra concepts, then introduce number theory concepts related to working mod n and present Pollard's $p-1$ factorization method. We will then establish a group structure on elliptic curves, which will allow us to build on key ideas from Pollard's method in order to understand Lenstra's elliptic curve method of factorization.

Contents

1. Introduction

The idea of this paper is for readers even with very little prior knowledge of any algebraic or number theory concepts to be exposed to and understand the theory behind a few important factorization methods, and also be introduced to a new mathematical concept in elliptic curves, and showcase their applications.

When finding prime factors of numbers, we know given some number n, if we check all prime numbers $\leq n$, we will have enough information to either find a factor, or determine that n is prime. However, when n becomes large, performing such a test is extremely inefficient, leading to the development of faster and more efficient factorization methods. These large n that are difficult to factor are core to different encryption methods being used today, and as a result finding ways to efficiently factor is important to keep our information safe, and develop stronger encryption methods.

2. Algebra Preliminaries

Before diving into factorization methods, it is important to establish some groundwork. Here we present algebraic concepts that will be especially relevant throughout this paper.

Definition 2.1. A **group** is defined as a set G with an operation, $\cdot : G \times G \to G$ that satisfies the following properties

1. (Closure) For all a, b in G, a $\cdot$ b is also in G
2. (Associativity) For all a, b, c in G, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$
3. (Identity) There exists an element e in G such that for all a in G, $a \cdot e = e \cdot a = a$
4. (Inverse) For each a in G, there exists some b in G such that $a \cdot b = b \cdot a = e$

Definition 2.2. An **abelian group** is a group that fulfills an additional fifth property.

5. (Commutativity) For all a, b in G, $a \cdot b = b \cdot a$

Definition 2.3. Given two groups F and G, with operations + and $\cdot$ respectively, a **group homomorphism** is a map between two groups, $f : G- > H$ such that:

1. $f(e_G) = e_H$
2. $f(g_1 + g_2) = f(g_1) \cdot f(g_2)$

An **isomorphism** is a bijective homomorphism.

3. Mod n and Factorization

We will begin our discussion by first talking about the greatest common denominator, or gcd, of two numbers: Given two positive integers a and b, we can find $gcd(a, b)$ through a procedure known as the Euclidean algorithm.

The procedure is as follows: Without loss of generality, assume $a > b$. Find a solution (q_1, r_1) for the equation $a = q_1 b + r_1$ where $0 \leq r_1 < b$ and $q_1 \geq 0$ from the restriction on r and from the fact that $a < b$. If $r_1 = 0$, b divides a and q_1 is our greatest common denominator. If not, continue by trying to find $gcd(b, r_1)$ and find solution to the equation $b = q_2 r_1 + r_2$, where $0 \leq r_2 < r_1$. If $r_2 \neq 0$, we continue, next solving for q_3 and r_3 where $r_1 = q_3 r_2 + r_3$. We continue similarly until we yield a solution to $r_{k-1} = q_{k+1} r_k + r_{k+1}$, where $r_{k+1} = 0$. A solution must be reached since each r_k is a non-negative integer, and we know $r_{k+1} < r_k < r_{k-1} ... < r_2 < r_1 < b$ from our restrictions on each r_k. This means there will always a value k where $r_{k+1} = 0$. Once this point is reached, $gcd(a, b) = r_k$. The reason this algorithm works is because $gcd(a, b) = gcd(a - b, b) = gcd(a - q_1 b, b) = gcd(r_1, b) = gcd(r_2, r_1) = ... = gcd(r_{k-1}, r_k)$.

Example 1. Let $a = 318$, $b = 120$. We use the Euclidean algorithm to find $gcd(a, b)$.

$$312 = 2 \cdot 120 + 78$$

$$120 = 1 \cdot 78 + 42$$

$$78 = 1 \cdot 42 + 36$$

$$42 = 1 \cdot 36 + 6$$

$$36 = 6 \cdot 6 + 0$$

$$gcd(318, 120) = 6$$

If we find that $gcd(a, b) = 1$, we say that a and b are relatively prime to each other. To further our discussion of relatively prime numbers, we will introduce Bezout's Identity [2].

Theorem 3.1. *(Bezout's Identity) Given two positive, relatively prime integers a, b, then there exist integers x and y that yield solutions to the equation*

$$ax + by = 1$$

Proof. We start by following the process of finding $gcd(a, b)$ using the Euclidean algorithm, and make a careful rearrangement at each step. We want to show any remainder r_i can be expressed in the form $ax_i + by_i$. From the first step of the algorithm, $a = q_1b + r_1$, and rearranging we see $r_1 = a - q_1b$. In this case, we have $x_1 = 1$ and $y_1 = -q_1$. Similarly from our second step, $b = q_2r_1 + r_2$, rearranging yields $r_2 = -q_2r_1 + b$. Substituting gives us $r_2 = -q_2(a - q_1b) + b = a(-q_2) + b(1 + q_1q_2)$. Here, $x_2 = -q_2$ and $y_2 = 1 + q_1q_2$. In general, to find solutions to the equation $r_i = ax_i + by_i$,

$$\begin{aligned} r_i &= ar_{i-2} + q_ir_{i-1} \\ &= ax_{i-2} + by_{i-2} - q_i(ax_{i-1} + by_{i-1}) \end{aligned}$$

which we can rearrange to find

$$r_i = a(x_{i-2} - q_ix_{i-1}) + b(y_{i-2} + q_iy_{i-1})$$

And since this holds true for all r_i, in particular $r_i = 1$ by virtue of $gcd(a, b) = 1$, we have a solution for our equation. □

Understanding the gcd and this property leads well into the topic of modular arithmetic. Modular arithmetic is a powerful tool that allows for increased efficiency in factorization algorithms and makes great use of the properties above.

Definition 3.2. Given integers a, b, and n, with $n \neq 0$, we say that $\mathbf{a \equiv b (mod\ n)}$ if $a - b$ is an integer multiple of n.

We call the set $[a]_n = \{z \in \mathbb{Z} | z = a + kn, k \in \mathbb{Z}\}$ the **congruence class** of a modulo n.

The notation $\mathbb{Z}/n\mathbb{Z}$ will be used in the future to refer to the set of congruence classes of integers mod n. We choose one representative per congruence class, and it is the convention to write $\mathbb{Z}/n\mathbb{Z} = \{0, 1, 2...n - 1\}$.

It is important to note that we can use $\mathbb{Z}/n\mathbb{Z}$ to form a group:

Theorem 3.3. *$\mathbb{Z}/n\mathbb{Z}$ forms an abelian group over addition modulo n, where addition modulo n is defined as $[a]_n + [b]_n = [a + b]_n$*

The proof for the above is left as an exercise for the reader. This group structure will be relevant to our use of operations over mod n, as well as our later applications over elliptic curves.

As a result, addition modulo n is well defined, and we can similarly perform subtraction modulo n through the use of additive inverses that exist as part of our group structure.

When looking at multiplication mod n, we must first establish what it means to be a multiplicative inverse mod n.

Definition 3.4. Given two integers a and b, we say they are **multiplicative inverses** of each other mod n if $ab \equiv 1 (mod\ n)$.

Theorem 3.5. *Given integers a, s, n, t with $gcd(a, n) = 1$ and $as + nt = 1$, then s is the multiplicative inverse of a.*

Proof. Rearranging quickly yields $as - 1 = -nt$, thus $as - 1$ is a multiple of n.
$as - 1 \equiv 0(mod\ n) \implies as \equiv 1(mod\ n)$. □

As a result of this property, if $gcd(a, n) \neq 1$, we do not have a multiplicative inverse mod n, and thus we cannot form a group over multiplication. However, we can still apply multiplication mod n in the form of repeated addition.

Using multiplication mod n, we also find that it is much easier to compute the value of numbers raised to large powers mod n. To showcase this, if we want to compute $2^5(mod\ 7)$, we notice $2^3(mod 7) \equiv 8(mod\ 7) \equiv 1(mod\ 7)$.

Thus $2^5(mod\ 7) \equiv (1 \cdot 4)(mod\ 7) \equiv 4(mod\ 7)$.

This technique of quickly calculating powers mod n is known as modular exponentiation.

Division is not well defined for *all* integers modulo n; however understanding when we are able to divide mod n and how it is defined in these cases is especially relevant to the factorization algorithms we will cover later in this paper. The following theorem gives us some insight into when we can apply division mod n.

Theorem 3.6. *Given integers a,b,c,n such that gcd(a,n) = 1 and $ab \equiv ac(mod\ n)$, then $b \equiv c(mod\ n)$.*

Proof. Since $gcd(a, n) = 1$, we can find x, y such that $ax+ny = 1$. We can multiply both sides by $(b - c)$ to yield $(ab - ac)x + (b - c)ny = (b - c)$.

From our statement $ab \equiv ac(mod\ n) \implies ab - ac \equiv 0(mod\ n)$, so $(ab - ac)$ is a multiple of n, $(b - c)ny$ is also a multiple of n, thus the right side must also be a multiple of n so $b \equiv c(mod\ n)$ □

Theorem 3.5 also tells us that integers relatively prime to n will always have multiplicative inverses mod n. This powerful fact allows us to deal with fractions mod n.

Given p, q, and n, where q and n are relatively prime, we note that $gcd(q, n) = 1$ implies that q has a multiplicative inverse mod n. This means we can think of $\frac{p}{q}(mod\ n)$ as $p \cdot q^{-1}(mod\ n)$.

Example 2. $\frac{2}{3}(mod\ 11) \equiv 2 \cdot 3^{-1}(mod\ 11) \equiv 2 \cdot 4(mod\ 11) \equiv 8(mod\ 11)$.

Modular arithmetic also yields two powerful theorems that will be used throughout this paper.

Theorem 3.7. *(Fermat's Little Theorem)[2] Given a prime number p, and integer a which is not divisible by p, then $a^{p-1} \equiv 1(mod\ p)$*

Proof. Consider set $S = 1, 2, 3...p - 1$ and function $f(x) = ax(mod\ p)$ defined for $x \in S$. We claim that the image of f is exactly S. Suppose for contradiction there exists some $x \in S$ such that $f(x)$ is not in S. Then $f(x) \equiv 0(mod\ p)$ so $ax \equiv 0(mod\ p)$. As a and p are relatively prime, we can divide both sides by a by Theorem 3.6. This gives us $x \equiv 0(mod\ p)$ contradicting our statement that $x \in S$. Therefore $f(x) \in S$ for all $x \in S$.

Consider x, y in S. Notice $f(x) = f(y) \implies ax \equiv ay(mod\ p)$, and similarly divide by a to find $x \equiv y(mod\ p)$. This gives us the fact that if $x \neq y$ then $f(x) \neq f(y)$, and therefore $f(1), f(2)...f(p - 1)$ are all distinct elements of S.

From here, it follows

$$\begin{aligned} 1 \cdot 2 \cdot 3... \cdot (p-1) &\equiv f(1) \cdot f(2)... \cdot f(p-1)(mod\ p) \\ &\equiv (a \cdot 1) \cdot (a \cdot 2)... \cdot (a \cdot (p-1))(mod\ p) \\ &\equiv a^{p-1} \cdot (1 \cdot 2... \cdot (p-1))(mod\ p) \end{aligned}$$

Again, by virtue of p being prime, $gcd(i, p) = 1$ for all $1 \leq i \leq p-1$, so we can divide both sides of the above equation by $(1 \cdot 2 \cdot 3... \cdot (p-1))$ to yield

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

□

Before we jump into the next important theorem, we introduce the definition of a **direct sum**, which will be relevant for this theorem.

Definition 3.8. The **direct sum**, $\bigoplus$, of two commutative groups, $(G_1, +_1)$ and $(G_2, +_2)$ is defined to be the set of ordered pairs formed from elements of G_1 and G_2

$$G_1 \bigoplus G_2 = \{(g_1, g_2) | g_1 \in G_1, g_2 \in G_2\}$$

These ordered pairs can be added component wise,

$$(g_{1_1}, g_{2_1}) + (g_{1_2}, g_{2_2}) = (g_{1_1} +_1 g_{1_2}, g_{2_1} +_2 g_{2_2})$$

meaning $G_1 \bigoplus G_2$ forms a group with the addition shown above and (e_{G_1}, e_{G_2}) as the identity element.

Theorem 3.9. *(Chinese Remainder Theorem) Let m and n be two relatively prime integers. If $x \equiv a(mod\ m)$ and $x \equiv b(mod\ n)$ then there is one unique solution for $x \equiv c(mod\ mn)$.*

Another way of expressing this result is that we have $\mathbb{Z}/(mn)\mathbb{Z} \simeq \mathbb{Z}/n\mathbb{Z} \bigoplus \mathbb{Z}/m\mathbb{Z}$.

Proof. (Existence) With m and n relatively prime, there are integers s and t such that $ms + nt = 1$ by Theorem 3.1. Rearranging, we find $ms = 1 - nt$ which implies $ms \equiv 1(mod\ n)$. Similarly $nt = 1(mod\ m)$ We claim if we have y such that $y = ant + bms(mod\ mn)$, then $x \equiv y(mod\ mn)$.

We notice $y \equiv ant(mod\ m) \equiv a(mod\ m)$. Additionally, $y \equiv bms(mod\ n) \equiv b(mod\ n)$, showing that y fulfills both equations from our theorem statement.

(Uniqueness) If we have another solution x_1, then $x \equiv x_1(mod\, m)$, $x \equiv x_1(mod\, n)$ meaning $x - x_1 \equiv 0(mod\ m)$ and $x - x_1 \equiv 0(mod\ n)$. Since $gcd(m, n) = 1$, $x - x_1 \equiv 0(mod\ mn)$ which implies $x \equiv x_1(mod\ mn)$.

(Isomorphism) We define map $f : \mathbb{Z}/mn\mathbb{Z} \to \mathbb{Z}/m\mathbb{Z} \bigoplus \mathbb{Z}/n\mathbb{Z}$. This map is a group homomorphism as $f(0) = (0, 0)$, giving us our identity property, and $f(x + y) = (x+y(mod\, n), x+y(mod\, m)) = (x(mod\, n), x(mod\, m))+(y(mod\, n), y(mod\, m)) = f(x) + f(y)$. From the uniqueness proved above, we have injectivity. From having mn combinations to map to with mn choices mod mn, again due to uniqueness we have surjectivity. As a result we have an isomorphism. □

Using these tools, we can now introduce Pollard's $p-1$ factorization method [2], which is as follows.

Definition 3.10. We define the $p-1$ method as follows: Given some large n to factor, we can choose some integer $a > 1$ and some bound B. We desire to compute $a^{B!}$ which we will call b. If $gcd(b-1, n) > 1$, we have a nontrivial factor of n.

To compute b, we set $b_1 \equiv a(mod\ n)$ and $b_j \equiv b_{j-1}^j(mod\ n)$, so that $b_B \equiv b(mod\ n)$.

Example 3. In order to attempt to factor 1403, we can pick $a = 2$, and start by selecting $B = 4$.

$$2^{4!} \equiv 142(mod\ 1403)$$

$$gcd(2^{4!} - 1, 1403) = gcd(141, 1403) = 1$$

For $B = 4$, the test failed, but if we try $B = 5$

$$2^{5!} \equiv (2^{4!})^5 \equiv 794(mod\ 1403)$$

$$gcd(2^{5!} - 1, 1403) = gcd(793, 1403) = 61$$

Through the algorithm, we have found a nontrivial factor of 1403.

The reason this method works is that if we have some prime factor p of n, where $n = pq$, and $p - 1$ is made up entirely of small primes, then as a result of only containing small primes there is a good chance $p - 1$ divides $B!$.

Thus, $b \equiv a^{B!} \equiv a^{(p-1)*k}(mod\ p)$ through $p - 1$ dividing $B!$. We can then apply Fermat's theorem to see $a^{(p-1)*k} \equiv a^{p-1} \equiv 1(mod\ p)$.

4. Introduction to Elliptic Curves

To understand the workings of the Lenstra's elliptic curve factorization method, we must first familiarize ourselves with elliptic curves.

For the sake of this paper, we will only be dealing with curves of characteristic > 3.

Definition 4.1. An **elliptic curve** over a field K, $E(K)$ is a curve typically of the form

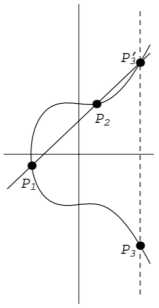
$$y^2 = x^3 + Ax + B$$

where $E(K) = \{(x, y) \in K \times K | y^2 = x^3 + Ax + B\} \cup \{O\}$ and $A, B \in K$. O is known as the point infinity, whose importance will be made clear shortly.

The above equation is referred to as the **Weierstrass equation**. From this point onward, we will be dealing with elliptic curves of this form.

On an elliptic curve $E(K)$, given two points P_1 and P_2 on the curve, we are able to formally define an addition between these two points.

For some visual intuition as to what addition would look like, if we have our elliptic curve E over the field of real numbers, $\mathbb{R}$, $P_1 + P_2$ can be seen through the image below ([1] pg. 12).



Here, $P_1 + P_2$ can be thought of as the reflection over the x-axis of the point of intersection between the line through P_1 and P_2, and our curve $E(\mathbb{R})$. We now present a rigorous definition for point addition on elliptic curves.

Definition 4.2. For an elliptic curve over field K, $E(K)$, for points $P, Q \in E(K)$, $P = (x_1, y_1)$, $Q = (x_2, y_2)$, we define $P + Q$ as the point $(x_3, y_3) \in K \times K \cup \{O\}$ where $x_3 = m^2 - x_1 - x_2$ and $y_3 = -y_1 + m(x_1 - x_3)$ where m is defined as follows:
If $P \neq Q$, $m = (y_1 - y_2)/(x_1 - x_2)$
If $P \neq Q$, but $x_1 = x_2$, $m = \infty$ and $P + Q = O$
If $P = Q$, $m = (3x_1^2 + A)/2y_1$
If $Q = O$, $P + O = O + P = P$

The source of these formulas can be thought of as follows: If $P \neq Q$, then we can draw a line between P and Q with slope $m = (y_1 - y_2)/(x_1 - x_2)$, yielding the line equation $y = m(x - x_1) + y_1$. Squaring, we get $y^2 = (m(x - x_1) + y_1)^2$

Setting this equal to our Weierstrass equation for elliptic curves, we find $(m(x - x_1) + y_1)^2 = x^3 + Ax + B$. Setting the left side to 0, we find

$$0 = x^3 - m^2x^2 + 2m(y_1 - mx_1)x + m^2x_1^2 + y_1^2 - 2mx_1y_1$$

Finding the roots of this cubic will yield the intersection points between our line and our curve, and we already know two solutions, x_1 and x_2, meaning we are searching for our third root, x_R.

Expanding the expression $(x - x_1)(x - x_2)(x - x_R)$ yields

$$x^3 - (x_1 + x_2 + x_R)x^2 + (x_1x_2 + x_1x_R + x_2x_R)x - x_1x_2x_R$$

Since this cubic must be equal to our prior cubic, we know

$$m^2x = -(x_1 + x_2 + x_R)x$$

Rearranging gives us $x_R = m^2 - x_1 - x_2$, since the x coordinate value does not change from flipping over the horizontal axis, thus $x_3 = x_R = m^2 - x_1 - x_2$.

Since slope holds constant over a line, we can express $m = (y_R - y_1)/(x_R - x_1)$, rearranging yielding $y_R = m(x_R - x_1) + y_1$. Flipping over the horizontal axis, we find $y_3 = -y_R = m(x_1 - x_3) - y_1$.

In the case where $P = Q$, we want to find the slope of the tangent line at (x_1, y_1), so we can perform implicit differentiation to find our slope.

$2y\frac{dy}{dx} = 3x^2 + A \implies \frac{dy}{dx} = \frac{3x^2+A}{2y}$, giving us our m.

Note that when $Q = O$, the line through P and O is vertical and intersects $E(K)$ at some point R which is the reflection of P over the x-axis. Reflecting back over the x-axis to find $P + O$ yields P. This holds true for all P, thus we have an identity element.

If $y_1 = 0$, we say $P + Q = O$.

With addition over elliptic curves defined, we can use this notion of addition to show that an elliptic curve $E(K)$ equipped with addition forms a group.

Theorem 4.3. *An elliptic curve over field K with point addition as defined above, $E(K)$ forms an abelian group.*

Proof. To show the existence of an abelian group, we must confirm the 5 properties.

Commutativity: We notice for any 2 points P, Q on $E(K)$, that the line through P and Q is the same as the line through Q and P. If $Q = O$, we know $P + Q = Q + P = P$.

Identity: From the properties presented earlier with our O element, we see for all points P in $E(K)$, $P + O = P$.

Inverse: Given point P, we can always find the inverse of P, P^{-1} by reflecting across the x-axis, so we have an inverse for all P in $E(K)$. $P + P^{-1} = O$.

Closure: From our definition of point addition, for points P, Q on $E(K)$, $P + Q$ must either be a point on the curve $y^2 = x^3 + Ax + B$ or the point at infinity, O, giving us closure.

Associativity: There are many methods of proving the associativity of elliptic curve addition, including projective spaces, Bezout's theorem, and computation of the formula themselves. As the proof using the formula derived above becomes messy and unwieldy, we will forego a formal proof of associativity for this paper. If you seek such a proof, refer to [1] pg. 21 - 25.

□

From this group structure on elliptic curves, we can make powerful statements about elliptic curves. Upon deeper exploration and application of these formulae alongside the behavior of elliptic curve as rings, both of which go beyond the scope of this paper, we can extend several useful theorems and ideas to elliptic curve over fields of the form $\mathbb{Z}/n\mathbb{Z}$.

In particular, we can extend the Chinese Remainder Theorem to elliptic curves in the following way:

Theorem 4.4. *Given odd integers m and n with $gcd(m, n) = 1$, let $y^2 = x^3+Ax+B$ be the equation for an elliptic curve E. Then, considering E over fields $\mathbb{Z}/mn\mathbb{Z}$, $\mathbb{Z}/m\mathbb{Z}$, and $\mathbb{Z}/n\mathbb{Z}$, we have a group isomorphism:*

$$E(\mathbb{Z}/mn\mathbb{Z}) \simeq E(\mathbb{Z}/m\mathbb{Z}) \bigoplus E(\mathbb{Z}/n\mathbb{Z}). \tag{4.5}$$

The reason we have these restrictions on m and n is similar to the reason we wanted the characteristic of curves not to be 2 or 3, as it causes issues to arise during computation.

For a proof of the above statement, refer to [1] pg. 67-70.

This extension of the Chinese remainder theorem will later aid in our understanding of how and why the elliptic curve factorization method works.

5. The Elliptic Curve Factorization Method

The idea behind this method is drawing a connection between finding an element that has no multiplicative inverse *modn* and repeated elliptic curve addition. Recall from Section 2 our discussion about computing fractions *modn* that we are unable to compute the value of a fraction *modn* if our denominator is non-invertible *modn*. Additionally, recall that as part of our addition algorithm over elliptic curves, we are required to calculate the value of our m term, taking the form of either $\frac{x_2-x_1}{y_2-y_1}$ or $\frac{3x^2+A}{2y}$, and these computations can only be completed *modn* if our $y_2 - y_1$ term or $2y$ term are relatively prime to n.

We will now follow the example provided in [1] to showcase how a single elliptic curve and point pair can find a non-trivial factor, and then introduce the algorithm.

Example 4. We desire to factor 4453. Let elliptic curve $E(\mathbb{Z}/4453\mathbb{Z})$ be defined by the equation $y^2 = x^3 + 10x - 2(mod\ 4453)$, and let point $P = (1, 3)$.

We first compute $2P$. The slope of our tangent line is equal to

$$\frac{3x^2 + A}{2y} \equiv \frac{13}{6} \equiv 3713(mod\ 4453)$$

as we can find $6^{-1} \equiv 3711(mod\ 4453)$. From here we compute $2P$

$$x \equiv 3713^2 - 2 \equiv 4332,\ y \equiv (-3713)(x-1) - 3 \equiv 3230$$

We now desire to compute $3P = 2P + P$. We first start by finding our slope:

$$\frac{3230 - 3}{4332 - 1} = \frac{3227}{4331}$$

But from here we find that 4331 has no multiplicative inverse mod 4453, because $gcd(4331, 4453) = 61 \neq 1$. Thus we have found a factorization for 4453, where $4453 = 61 \times 73$

Using this power of some elliptic curve mod n being able to discern a non-trivial factor for n, we present Lenstra's elliptic curve algorithm [3].

Definition 5.1. Lenstra's elliptic curve factorization algorithm: To attempt to factor some n, we begin by picking some search limit, C, and generate a set of around 15 random elliptic curves $E_i(\mathbb{Z}/n\mathbb{Z}) : y^2 = x^3 + A_i x + B_i$ alongside some point $P_i = (x_i, y_i) \in E_i(\mathbb{Z}/n\mathbb{Z})$.

A method for generating this set of curves is as follows: For each i, choose a random integer A_i and random pair of integers $P_i = (u_i, v_i)$ mod n. Then compute $B_i = v_i^2 - u_i^3 - A_i u_i$, giving us elliptic curve with equation $y^2 = x^3 + A_i x + B_i$ and point (u_i, v_i).

From here we attempt to compute $C! * P_i$ for each pair of point and curve. If the computation fails for some P_i, we have found a non-trivial factor of n. If this step yields n as the non-trivial factor, generate a new curve and point. If the computation is successful (i.e. we find the point $C! * P_i$) we similarly generate a new curve and point and start over.

We can think of the multiplication required, $C! * P$, as repeated point addition. This means that we never need to compute the value of $C!$, as $2!P = P + P$, $3!P = 2P + 2P + 2P$, and we can similarly compute $C!P$ in this way.

Since we are multiplying the point P with many small prime factors during the process of generating $C! * P$, this draws a close similarity to Pollard's algorithm from section 3. Recall that Pollard's $p - 1$ algorithm was most effective when one of our desired prime factors, p, had $p - 1$ be composed of small primes. In this sense, running this factorization test on a single elliptic curve and point pair is equivalent to Pollard's method. However, when the $p - 1$ method fails, there is no next step, while with Lenstra's algorithm, we can carry out computations on multiple curves simultaneously, and generate new curves. As a result of this, we have a more efficient approach for factoring n along with a higher likelihood of success.

In order to provide deeper insight as to why this algorithm is able to work efficiently, we will introduce the notion of "smoothness" and Hasse's theorem (Hasse, 1936).

Definition 5.2. We say some integer n is **C-smooth** if all the prime factors of n are less than or equal to C.

Theorem 5.3. ***Hasse's Theorem****: Given an elliptic curve E over a finite field $\mathbb{F}_q$, where q is a power of a prime number, then the order of $E(\mathbb{F}_q)$, $\#E(\mathbb{F}_q)$, satisfies the following relation:*

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

For a proof of Hasse's theorem, refer to [1] pg. 98 - 100. In fact, each integer in this interval occurs as the order of some elliptic curve over $\mathbb{F}_q$.

If $n = pq$, from Theorem 4.4, we know

$$E(\mathbb{Z}/n\mathbb{Z}) \simeq E(\mathbb{Z}/p\mathbb{Z}) \bigoplus E(\mathbb{Z}/q\mathbb{Z})$$

This means that in attempting to factor our n, for each elliptic curve we generate over $\mathbb{Z}/n\mathbb{Z}$, we are also implicitly generating curves over $\mathbb{Z}/p\mathbb{Z}$ and $\mathbb{Z}/q\mathbb{Z}$ at the same time, even if we do not know the values of p and q.

This fact allows us to apply Hasse's theorem, where we know that the order of points of an elliptic curve over field $\mathbb{F}_p$ is in the range $[p+1-2\sqrt{p}, p+1+2\sqrt{p}]$. And in fact, each integer in this interval occurs for *some* elliptic curve mod n, meaning if we generate enough curves, their orders will be more uniformly distributed.

In particular, we are looking for integers that are C-smooth in our Hasse interval. As we increase the size of our C, we also increase the density of C-smooth integers in the Hasse interval (Note: in the $p-1$ algorithm, we relied on the possibility of $p-1$ being C-smooth). As a result, in generating pairs of curves and points, we increase the chance of the order of one of these curves, E_i, being C-smooth, and in turn make it more likely for one of these curves to yield us a non-trivial factor of n.

However, as we increase the size of our bound C, we also decrease the efficiency of our algorithm, as it increases the average amount of computations required per curve. Each curve computation requires storage, and since we are doing point addition, requiring data for our x and y, this is more costly than only storing data for one variable. As a result of this, the algorithm is best suited for factoring numbers up to 40 to 50 digits long. If we go past this point, the bound C necessary to generate a high enough density of C-smooth numbers in the Hasse interval slows down the algorithm to where there are better alternatives.

6. Conclusion and Further Readings

Modern methods of factorization have greatly improved upon earlier methods, as seen in the difference in efficiency between the $p-1$ method and the elliptic curve method, and as a result there are powerful tools at our disposal to try and find solutions to the difficult problems we face today related to prime numbers, cryptography, and encryption. Due to Lenstra's algorithm not being particularly efficient for extremely large n, it is used more as an intermediate step in modern day factorization attempts. For the reader interested in further learning about factorization methods, both old and new, [2] chapters 6, 7, and 16 offer in-depth looks into different factorization algorithms and why they work. In particular, these chapters discuss the quadratic sieve method, which is a core idea in today's fastest factorization algorithms. For those interested in learning more about elliptic curves and their many applications, [1] offers a thorough algebraic establishment of the properties of elliptic curves, with chapters 5, 6, and 7 providing great information about their applications to cryptography and primality testing.

Acknowledgements

I would like to give a huge thanks to my mentor, Maeve Coates Welsh, for both inspiring the topic of this paper, and helping me learn and put together all the information presented here. Constantly providing me with amazing resources, and taking the time to research and explain gaps in my knowledge, this paper would not have been possible without her. I would also like to thank Peter May for putting together the REU even during this pandemic and allowing me this amazing opportunity.

References

[1] Lawrence C. Washington Elliptic Curve Number Theory and Cryptography Taylor & Francis Group. 2008

[2] Wade Trappe & Lawrence C. Washington Introduction to Cryptography With Coding Theory Pearson Education Inc. 2002.

[3] H.W. Lenstra Factoring Integers with Elliptic Curves Math Department, Princeton University, 1987

[4] Mirjam Soeten Hasse's Theorem on Elliptic Curves University of Groningen. 2013.