ELLIPTIC CURVES AND LENSTRA'S FACTORIZATION ALGORITHM

DANIEL PARKER

Abstract. This paper provides a foundation for understanding Lenstra's Elliptic Curve Algorithm for factoring large numbers. We give a definition of elliptic curves over fields of characteristic not 2 or 3, followed by a construction of the abelian group over the K-rational points of an elliptic curve. Next, Pollard's $p-1$ algorithm is explained, as well as the Hasse-Weil Bound, after which follows a discussion of how Lenstra's Algorithm improves upon Pollard's. Then Lenstra's Algorithm is explained in full, followed by a brief note on its application.

Contents

1. Introduction

The study of elliptic curves encapsulates a unique intersection of algebra, geometry, and number theory. This paper concerns Lenstra's Algorithm for factoring large numbers, which is a perfect example of how these fields intersect. Before discussing the algorithm itself, we introduce elliptic curves and the group structure on which Lenstra's Algorithm depends, and also contextualize the algorithm with its predecessor, Pollard's $p-1$ Algorithm.

We assume some background in projective geometry, but this is not essential to understanding the paper holistically. Those interested in a good introduction to the relevant parts of projective geometry, or who are seeking further information about elliptic curves in general, are encouraged to consult [2].

Date: August 2014.

2. Elliptic Curves

Here we define elliptic curves over a field K such that K does not have characteristic 2 or 3. Curves over fields with characteristic 2 or 3 have longer general equations that complicate their eventual use in Lenstra's Algorithm.

Definition 2.1. Let K be a field either of characteristic 0 or characteristic greater than 3. Then we define an elliptic curve E over K to be the projective closure of a nonsingular curve over K of the form

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

where $A, B \in K$.

To describe this projective closure, we set $f(x, y) = y^2 - x^3 - Ax - B$, then consider the homogenization of $f(x, y)$:

$$F(X, Y, Z) = Z^3 f(\frac{X}{Z}, \frac{Y}{Z}) = Y^2 Z - X^3 - AXZ^2 - BZ^3.$$

The projective closure is the set of solutions to $F(X, Y, Z) = 0$. Since in projective space any point $(X : Y : Z)$ is equivalent to $(\lambda X : \lambda Y : \lambda Z)$ where $\lambda \in K$ is nonzero, the solutions where $Z \neq 0$ are equivalent to the solutions where $Z = 1$: the affine solutions $f(x, y) = 0$.

Now we consider when $Z = 0$. Plugging in, we get $X^3 = 0$, indicating that the only remaining solution is $(0 : 1 : 0)$. We will call this the curve's point at infinity, referred to with the letter O, and the rest of the points will be referred to by their corresponding affine point.

Before we move on, we give one more definition.

Definition 2.2. We define the set of K-rational points of an elliptic curve E as the set of points on E whose coordinates all lie in K, as well as the point O, and denote it $E(K)$.

3. The Group Law

We now outline the construction of an abelian group structure on the K-rational points of an elliptic curve E over a field K. To accomplish this, we first give such an abelian group structure over the algebraic closure $\overline{K}$. We then show that $E(K)$ is a subgroup of $E(\overline{K})$, which implies that $E(K)$ itself is an abelian group.

Let E be an elliptic curve over a field K of either characteristic 0 or characteristic greater than 3, and let $\overline{K}$ be the algebraic closure of K. Since $K \subset \overline{K}$, we can also consider E over $\overline{K}$. Let L be any line in projective space, let S be any $\overline{K}$-rational point, and let m_S denote the multiplicity of the intersection of E and L at S. As $\overline{K}$ is algebraically closed, and $\deg(E) = 3$, the total multiplicity over all points is 3, i.e., $\sum_S m_S = 3$. Without loss of generality, and bearing in mind that due to multiplicity, any two or all three points could be equal, we refer to the three points of intersection of L with E as P, Q, and R. Also, for any point $S \in E(\overline{K})$ with $S = (X_S : Y_S : Z_S)$, we define $-S$ to be the point $(X_S : -Y_S : Z_S)$. Note that by the definition of an elliptic curve, $S \in E(\overline{K})$ implies that $-S \in E(\overline{K})$. Furthermore, since $(0 : 1 : 0) = (0 : -1 : 0)$, we can see that $-O = O$. With these facts in mind, we define addition over $E(\overline{K})$ by saying that if $P, Q, R \in E(\overline{K})$ are collinear, then

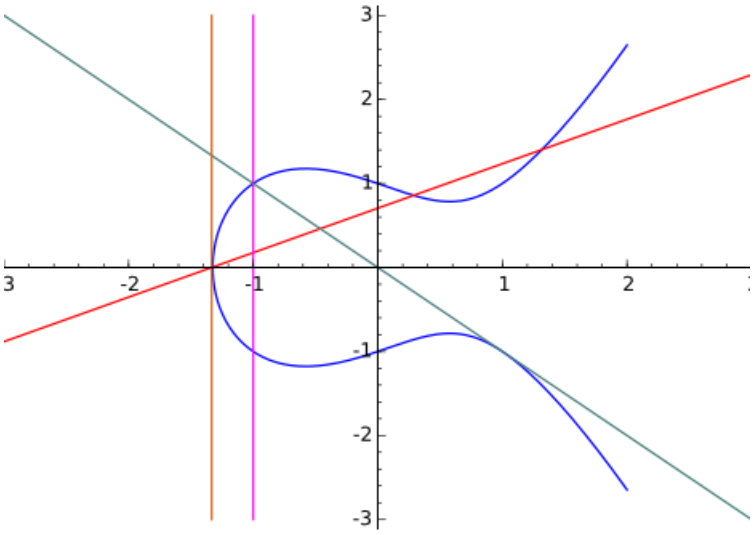

FIGURE 1. Some potential intersections of a line with an elliptic curve.

$$P + Q = -R.$$

To find the sum of two points, we examine the line L between them, find the third point of intersection of L with E, and then invert that point. Now we show that this definition of addition constitutes an abelian group.

First of all, consider the intersection of the line $Z = 0$ with an elliptic curve E over $\overline{K}$ given by the equation $Y^2Z = X^3 + AXZ^2 + BZ^3$. As touched on before, the point of intersection O satisfies $X^3 = 0$. But now notice that this intersection has multiplicity 3. Therefore, $O + O = -O$, i.e., $O + O = O$.

Addition over $E(\overline{K})$ must be closed, since we already determined that any P, Q, R are all in $E(\overline{K})$ and that $S \in E(\overline{K})$ implies that $-S \in E(\overline{K})$.

Note also that it must be the case that for any $P, Q \in E(\overline{K})$, $P + Q = Q + P$, since P and Q determine exactly one line. Therefore, the operation is commutative.

Next, we show that for a point $P \in E(\overline{K})$, the additive inverse is indeed $-P$. Let $P = (x_P, y_P)$ where $x_P, y_P \in \overline{K}$. Then the only line intersecting both P and $-P$ is $X = x_P Z$. The third point at which this line intersects E is O, so $P + (-P) = -O = O$.

Now we show that O is the additive identity of addition over $E(\overline{K})$. Suppose we want to perform $O + P$. As determined earlier, the line between O and P is the vertical line $X = x_P Z$, making the third point of intersection $-P$. Therefore, $O + P = -(-P)$. From our definition of negation, $-(-P) = P$, and recalling commutativity, we then see that $O + P = P + O = P$.

Thus, we have demonstrated all conditions for $E(\overline{K})$ to be an abelian group except for associativity. We will not prove associativity here, as the proof is lengthy and is not constructive to understanding Lenstra's Algorithm, but one can verify associativity using the formulae given here and checking all possible cases.

From this characterization, we derive explicit formulae to calculate the sum of any two points. Let $P, Q \in E(\overline{K})$ such that $P = (x_P, y_P)$ and $Q = (x_Q, y_Q)$. First, we assume that $P \neq Q$. If $x_P \neq x_Q$, then we can easily find λ, the slope of the line between them:

$$\lambda = \frac{y_Q - y_P}{x_Q - x_P}$$

Next, we calculate the y-intercept of the line, ν, as

$$\nu = y_P - \lambda x_P = y_Q - \lambda x_Q.$$

Now that we have the equation for the line, we can substitute $(\lambda x + \nu)$ in for y in our elliptic curve:

$$y^2 = (\lambda x + \nu)^2 = x^3 + Ax + B$$
$$0 = x^3 - \lambda^2 x^2 + (A - 2\lambda\nu)x + (B - \nu^2)$$

By the fundamental theorem of algebra, and since $\overline{K}$ is algebraically closed, this cubic equation must have a third root (up to multiplicity), $x_R \in \overline{K}$, so

$$0 = x^3 - \lambda^2 x^2 + (A - 2\lambda\nu)x + (B - \nu^2) = (x - x_P)(x - x_Q)(x - x_R).$$

When we expand the right-most expression, the coefficient of the x^2 term is $-x_P - x_Q - x_R$. Therefore,

$$\lambda^2 = x_P + x_Q + x_R.$$

That means that we can isolate x_R in the equation, and subsequently find y_R:

$$x_R = \lambda^2 - x_P - x_Q$$
$$y_R = \lambda x_R + \nu.$$

As P, Q, and R are collinear, recall that for $R = (x_R, y_R)$, we have that $P + Q = -R = (x_R, -y_R)$. If x_P, x_Q, x_R are all distinct, the line connecting P, Q, and R will resemble the red line in Figure 1. If $x_R = x_P$ or $x_R = x_Q$, then the line will resemble the turquoise line in Figure 1.

Now suppose, contrary to our earlier assumption, that $x_P = x_Q$. If $P \neq Q$, $y_Q = -y_P$ and the line connecting them is simply the vertical line $x = x_P$, so $Q = -P$, and $P + Q = O$. This case resembles the pink line in Figure 1.

If $P = Q$, then the line with multiplicity 2 at P is the tangent line to E at P, so to find the slope λ of the tangent line at P, we perform implicit differentiation:

$$y^2 = x^3 + Ax + B$$
$$2y\frac{dy}{dx} = 3x^2 + A$$
$$\frac{dy}{dx} = \frac{3x^2 + A}{2y}$$
$$\lambda = \frac{3x_P^2 + A}{2y_P}$$

We then proceed as before with the prior formulae and arrive at the same equations for x_R and y_R, unless $y = 0$. If $y = 0$, we again have a vertical line, so the

third point of intersection is again O. Thus, $P + P = O$. Note that in this case, $P = -P$. For $y \neq 0$, the $P = Q$ case resembles the turquoise line in Figure 1. If $y = 0$, the $P = Q$ case resembles the orange line.

Suppose that $P, Q \in E(K)$, and $P + Q = -R$. Assume $R \neq O$. Then using our formulae for x_R and y_R, in all cases, x_R and y_R are equal to arithmetic expressions of x_P, x_Q, y_P, y_Q, and A, all of which are in K, meaning that $x_R, y_R \in K$. If $R = O$, then by definition $R \in E(K)$. Since $R \in E(K)$, $-R \in E(K)$, so $E(K)$ is closed under addition of points. That makes $E(K)$ a subgroup of $E(\overline{K})$. Consequently, $E(K)$ is an abelian group.

Thus, we have characterized the abelian group $E(K)$ over a field K. We now discuss a few more relevant topics, then proceed to Lenstra's Algorithm.

4. Pollard's p-1 Algorithm

First, we explain Pollard's Algorithm, because Lenstra's Algorithm is fundamentally an improvement of Pollard's. We begin with Fermat's Little Theorem.

Theorem 4.1. *Fermat's Little Theorem: If p is a prime number and $a \in \mathbb{N}$, then*

$$a^p \equiv a \mod p.$$

Further, if $p \nmid a$,

$$a^{p-1} \equiv 1 \mod p.$$

From this, we can see that raising both sides to some other power m preserves this relationship:

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

Suppose we have a composite number n, which we would like to factor. We know by the fundamental theorem of arithmetic that n has a prime factor $p \leq \sqrt{n}$. Thus, by Fermat's Little Theorem, if a is any integer, $a^{p-1} - 1 \equiv 0 \mod p$, so $p|(a^{p-1} - 1)$, and consequently, $p|(a^{(p-1)m} - 1)$. Therefore, $p| \gcd(a^{(p-1)m} - 1, n)$. If we could calculate $\gcd(a^{(p-1)m} - 1, n)$, we could find either p or some multiple of it. However, if we don't know what p is, then of course we don't know what $p - 1$ is either. But for some primes, this difficulty can be ameliorated.

Suppose that $p - 1$ is the product of small primes to small powers. If this is the case, and we have some k that is the product of many small primes to small powers such that $k = (p - 1)m$ for some m, then when we calculate $\gcd(a^k - 1, n)$, we will find p.

Thus we have the beginnings of a procedure for looking for prime factors of n. For $k = \text{lcm}(1, \cdots, K), K \leq \sqrt{n}$, and $1 < a < n$, we can calculate $\gcd(a^k - 1, n)$, and if $p - 1|k$, we will find p. If we don't find p, then we can increase K and keep looking. We are guaranteed to find p eventually, since when $K = \frac{1}{2}(p - 1)$, it must be the case that $p - 1|k$. But one can see how Pollard's Algorithm will do much better if $p - 1$ is the product of small primes.

If $p|a$, we can't use Fermat's Little Theorem this way. For this reason, we first check that $\gcd(a, n) = 1$. If not, then we have already found a factor!

Having discussed the reasoning behind the algorithm, here is an explicit set of steps.

- For some integer $K \leq \sqrt{n}$, let $k = \text{lcm}(1, \ldots, K)$.
- Choose a so that $1 < a < n$.
- If $\gcd(a, n) \neq 1$, then stop, as $\gcd(a, n)$ is a factor of n. Otherwise, continue.
- Calculate $D = \gcd(a^k - 1, n)$. If $1 < D < n$, then D is a factor of n. If $D = n$, then reselect a. If $D = 1$, then increase K.

The problem arises when n has no convenient factor p such that $p - 1$ is the product of small primes to small powers. In this case, k will not help us find the factor. Therefore, we have to continue increasing K until we reach, in the very worst case, $\frac{1}{2}(p - 1)$. We will see how this difficulty is resolved with Lenstra's Algorithm soon.

5. The Hasse-Weil Bound

Having discussed Pollard's algorithm, we now consider the Hasse-Weil Bound.

Theorem 5.1. *The Hasse-Weil Bound: Let p be prime, and $\#E(F_p)$ denote the number of F_p-rational points of an elliptic curve E over F_p. Then*

$$p + 1 - 2\sqrt{p} \leq \#E(F_P) \leq p + 1 + 2\sqrt{p}.$$

The proof of the Hasse-Weil bound is long and incorporates topics outside the scope of this paper, so it will not be covered. However, it will be useful in developing an analogous form of Pollard's algorithm for elliptic curves.

6. Lenstra's Algorithm

We said before that the problem with Pollard's Algorithm is that when there are no "good" factors of n, the algorithm's advantage fades considerably. Using Pollard's algorithm, the nonzero elements of $\mathbb{Z}/p\mathbb{Z}$ form a group of cardinality $p - 1$. If $p - 1 | k$, then $a^k \equiv 1 \mod p$. If $p - 1 \not| k$ then we have to increase k and hope that $p - 1 | k$ the next time, and we have bad performance when no "good" factors of n exist.

Using Lenstra's Algorithm, we substitute the nonzero elements of $\mathbb{Z}/p\mathbb{Z}$ with the $\mathbb{F}_p$-rational points of some elliptic curve E over $\mathbb{F}_p$. Instead of raising a random number a to a certain power k, we take a multiple kP of a point $P \in E(\mathbb{F}_p)$. Like in Pollard's Algorithm, when $\#E(\mathbb{F}_p)$ divides k, then $kP = O$, which will provide a factor of n. The difference is that when we "miss", by not having $\#E(\mathbb{F}_p)$ divide k, instead of increasing k, we can pick a random different curve E', so that $\#E'(\mathbb{F}_p)$ is different. We assume the conjecture (admittedly unproven, but very likely and backed by strong evidence) that each different curve's number of $\mathbb{F}_p$-rational points varies nearly randomly in the Hasse-Weil bound, so as we change the curve, it is rather likely that we will find some value of $\#E(\mathbb{F}_p)$ that divides k.

In order to see how this gives a factor of n, we must first discuss how we will calculate multiples of P. In fact, we will be doing so as if $\mathbb{Z}/n\mathbb{Z}$ were the field over which we are working, which seems problematic. It is true that $\mathbb{Z}/n\mathbb{Z}$ is not a field, as those elements $g \in \mathbb{Z}/n\mathbb{Z}$ where $\gcd(g, n) \neq 1$ lack multiplicative inverses. But when we are hunting for factors of n, this is just what we are looking for! Recall the formulae for x_R and y_R when $P + Q = -R$, derived in Section 3:

$$x_R = \lambda^2 - x_P - x_Q$$
$$y_R = \lambda x_R + \nu$$

For $x_P \neq x_Q$, we had

$$\lambda = (y_Q - y_P)(x_Q - x_P)^{-1},$$

making $x_Q - x_P$ the value that may not have an inverse. When $x_Q = x_P$, we instead use the other formula for λ

$$\lambda = (3x_P^2 + A)(2y_P)^{-1}$$

in the equations for x_R and y_R to find the sum of P and Q, and it will instead be $2y_P$ that may lack an inverse. We will call this value, situationally either $x_Q - x_P$ or $2y$, that may lack an inverse d.

When d lacks an inverse in $\mathbb{Z}/n\mathbb{Z}$, that means that $D = \gcd(d, n) > 1$. If $D < n$, then $D|n$. This is the mechanism by which we will find a factor of n. But when will d lack an inverse?

When $\#E(\mathbb{F}_p)|k$, since every $P \in E(\mathbb{F}_p)$ has order dividing $\#E(\mathbb{F}_p)$, $kP = O$ for any point on E when reduced mod p. This means that p divides the denominator d of kP. So as $p|d$, and p divides n, then $p|\gcd(d, n)$. Therefore, we will have this lack of an inverse exactly when $\#E(\mathbb{F}_p)|k$. Since k is a product of small primes to small powers, we will still be better at finding low primes than higher ones.

Some restrictions exist that must be addressed. As discussed before, the characteristic of the space over which we use our group formulae must not be 2 or 3, so we need to check that $2, 3 \nmid n$ before we begin. Also, if there exist m, r such that $m^r = n$, we will not find the factor m using this method, so we need to check that the roots of n, starting from the square root and stopping once a root evaluates to be less than 2, are not natural numbers.

With this explanation of the basic method done, now come the specific steps of the algorithm.

We want to find a factor for the composite integer $n \geq 2$.

- Check that $2, 3 \nmid n$ and that there do not exist m, r such that $m^r = n$.
- Choose random A, x_P, y_P such that $1 < A, x_P, y_P < n$.
- Let $B = y_1^2 - x_1^3 - bx_1 \mod n$. Then E is the curve $y^2 = x^3 + Ax + B$. Note that $P = (x_P, y_P)$ is on E. We will think of E as being an elliptic curve over $\mathbb{Z}/n\mathbb{Z}$, although $\mathbb{Z}/n\mathbb{Z}$ is not a field.
- Ensure that the curve is nonsingular on $\mathbb{Z}/n\mathbb{Z}$ by seeing that $\gcd(4A^3 + 27B^2, n) = 1$. If it equals n, then choose a different b. If it is between 1 and n, then we are done.
- Choose a number K and let k be $\mathrm{lcm}(1, \ldots, K)$.
- Attempt to compute kP over $\mathbb{Z}/n\mathbb{Z}$. If the formula does not fail at any point, then either try a new curve, or, after sufficient attempts with the same k, increase k. If this does fail, then we will have found a d that lacks an inverse in $\mathbb{Z}/n\mathbb{Z}$. If $\gcd(d, n) < n$, then we have factored n. If $d = n$, reduce k and try again.

The process of computing kP requires further explanation. To successively add P to itself requires many computations. Instead, we can use repeated doubling and k's binary expansion to find kP with fewer additions.

First we find k_0 through k_r where $k_i \in \{0, 1\}$ such that

$$k = k_o 2^0 + k_1 2^1 + k_2 2^2 + \cdots + k_r 2^r.$$

Next, we find $P_1 = P + P$, $P_2 = P_1 + P_1$, $P_3 = P_2 + P_2$, etc. Finally, we add

$$kP = k_0 P + k_1 P_1 + \cdots + k_r P_r$$

to find kP.

7. Applications

Lenstra's Elliptic Curve Algorithm, with optimizations, is the third fastest method for factoring large numbers, and the best for finding low factors. Programs for factoring large numbers will often begin with Lenstra's Algorithm, eventually switching to faster methods like the quadratic sieve. The continued use of this algorithm is a testament to the surprising depth and power of elliptic curves.

Acknowledgments. I would like to sincerely thank my mentor, Drew Moore. Drew was very willing to make himself available to meet with me throughout the program and has provided excellent guidance and knew when to push me to explain myself. He was also very patient with me, for which I am very grateful. I would also like to thank everyone who helped organize and taught in the 2014 UChicago Math REU, especially Laszlo Babai, who taught the Apprentice Program. Professor Babai encouraged our mathematical curiousity and challenged us to engage fully with the material. I would also like to thank my Mom and Dad for their support, and for paying for my train tickets to and from Chicago!

References

[1] Bjorn Poonen. Elliptic Curves. http://www-math.mit.edu/ poonen/papers/elliptic.pdf
[2] Joseph H. Silverman and John Tate. Rational Points on Elliptic Curves. Springer-Verlag New York, Inc. 1992.