ABSTRACT

Title of dissertation: CURVES AND THEIR APPLICATIONS TO FACTORING POLYNOMIALS

Enver Ozdemir, Doctor of Philosophy, 2009

Dissertation directed by: Professor Lawrence C. Washington
Department of Mathematics

We present new methods for computing square roots and factorization of polynomials over finite fields. We also describe a method for computing in the Jacobian of a singular hyperelliptic curve.

There is a compact representation of an element in the Jacobian of a smooth hyperelliptic curve over any field. This compact representation leads an efficient method for computing in Jacobians which is called Cantor's Algorithm. In one part of the dissertation, we show that an extension of this compact representation and Cantor's Algorithm is possible for singular hyperelliptic curves. This extension lead to the use of singular hyperelliptic curves for factorization of polynomials and computing square roots in finite fields.

Our study shows that computing the square root of a number mod p is equivalent to finding any of the particular group elements in the Jacobian of a certain singular hyperelliptic curve. This is also true in the case of polynomial factorizations. Therefore the efficiency of our algorithms depends on only the efficiency of the algorithms for computing in the Jacobian of a singular hyperelliptic curve. The algorithms for computing in Jacobians of hyperelliptic curves are very fast especially for small genus and this makes our algorithms especially computing square roots algorithms competitive with the other well-known algorithms.

In this work we also investigate superelliptic curves for factorization of polynomials.

CURVES AND THEIR APPLICATIONS TO FACTORING POLYNOMIALS

by

Enver Ozdemir

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2009

Advisory Committee:
Dr. Lawrence C. Washington, Chair
Dr. William Gasarch
Dr. Kartik Prasanna
Dr. Niranjan Ramachandran
Dr. Harry Tamvakis

Acknowledgments

First of all, I want to thank my advisor, Professor Larry Washington, for his invaluable guidance during my graduate study at University of Maryland. I am grateful for his willingness to help me with my research and his invaluable feedback for paper drafts. It has been a really pleasure to work with such an extraordinary advisor.

I would also like to thank my teachers at University of Maryland, Dr.Niranjan Ramachandran and Dr.Tom Haines, for inspiring me during my mathematical study. Prof.Prasanna, Prof.Tamvakis, and Prof.Gasarch deserve a special thanks as my thesis committee members.

I also want to thank Dr.Ayse Berkman for supporting me to become as a researcher in mathematics during my undergraduate study.

I would like to thank to my parents and my brother and sisters for their support and encouragement during my study.

Lastly I want to thank my wife, Zehra, for her support throughout last six years.

Table of Contents

Chapter 1

Background

1.1 Singular Curves

The methods that we offer for factorization of polynomials and computing square roots in finite fields heavily depend on the representations of the elements in Jacobians of singular curves and especially singular hyperelliptic curves. In this section we present a brief summary about singular curves and their Picard groups. More details about singular curves can be found in [16].

A *curve* over a field k is a reduced Noetherian connected scheme of dimension 1. A complete irreducible curve X is called *hyperelliptic* if there is a morphism $h : X \rightarrow \mathbb{P}^1$ of degree 2. The *Jacobian*, *Jac(X)*, of a hyperelliptic curve X is isomporhic to the identity component, *Pic^o(X)*, of the group Pic(X) which is the free abelian group of divisor(Cartier) classes of X modulo principal divisors. The curves that we use in the following chapters will be over finite fields and to utilize the equality Jac(X)=Pic$^o(X)$, we always assume that a curve over a finite field $\mathbb{F}_q$ has at least one $\mathbb{F}_q$-point.

In 1982, Mumford [17] presented a method for compact representation of an element in the Jacobian of a non-singular hyperelliptic curve. This representation can be considered as a nice application of the Riemann-Roch theorem and the theory of Weil divisors on non-singular curves. We describe the extension of this compact representation for singular hyperelliptic curves in chapter 2. This extension is also based on the Riemann-Roch theorem for singular hyperelliptic curves.

1.1.1 The Riemann-Roch Theorem for Singular Curves

Definition 1.1.1. Let X be an integral projective curve over a field k and D be a Cartier divisor in the group $\mathrm{Div}(X)$ of the Cartier divisors of X. We define:

$$\mathfrak{L}(D) = \{f \in k(X)^* \quad | \quad \mathrm{div}(f) + D \geq 0 \quad \} \cup \{0\}$$

$$l(D) = \dim_k \mathfrak{L}(D)$$

Theorem 1.1.2. *The Riemann-Roch theorem (first form)*

Let X be a complete integral projective curve over k and D be in Div(X). Then

$$l(D) \geq degD + 1 - g \tag{1.1}$$

where g is the arithmetic genus of X.

Proof. See [21, Section 4.3] or [16, Corollary 7.3.23] □

Let $\mathcal{O}_{X,P}$ be the local ring at the point $P \in X$ and $\widetilde{\mathcal{O}}_{X,P}$ be the integral closure of $\mathcal{O}_{X,P}$ in $k(X)$ where $k(X)$ is the function field of the curve X. Let P be in X. Then

$$\delta_P = \dim \left(\widetilde{\mathcal{O}}_{X,P}/\mathcal{O}_{X,P}\right)$$

is called the degree of singularity at P and by theorem 1 in [18] $\delta_P < \infty$. Let $\mathfrak{c}_P = \{f \in k(X) | f\widetilde{\mathcal{O}}_{X,P} \subset \mathcal{O}_{X,P}\}$ be the *conductor* of $\widetilde{\mathcal{O}}_{X,P}$ into $\mathcal{O}_{X,P}$. Since

$\delta_P < \infty$, we have $\mathfrak{c}_P \neq 0$ and we define $\deg(\mathfrak{c}_P) = \dim \left(\widetilde{\mathcal{O}}_{X,P}/\mathfrak{c}_P\right)$.

Theorem 1.1.3. *The Riemann-Roch theorem (definitive form)*

Let K be a canonical divisor and D be a divisor in Div(X). Then

$$l(D) - l(K - D) = degD + 1 - g \tag{1.2}$$

if and only if $deg(\mathfrak{c}_P) = 2\delta_P$ for all points $P \in X$. That is the equality holds if and only if the curve X is Gorenstein.

Proof. See [21, Chapter 4] or [18]. □

The above theorem is the main ingredient to give sufficient conditions for a certain divisor D in the Jacobian of a singular hyperelliptic curve to be a unique representative of its class.

1.1.2 Picard Groups of Singular Curves

Let X be a singular curve over the field k. The curves that we use for computational problems have singularities from self-intersection. As mentioned above, the compact representations of the group elements of $\mathrm{Pic}^o(X)$ is the main ingredient for our algorithms. We have sufficient tools to describe the Picard group of a non-singular hyperelliptic curve and to describe the elements of it and we essentially use the Picard group of a smooth curve to describe the Picard group of a singular curve. In order to do this, we need to parametrize the singular curve X by a non-singular curve X' and this operation is called *resolution of singularities* of X. More explicitly the resolution of singularities means find a smooth curve X' such that there exists a surjective proper morphism $\pi : X' \to X$. Fortunately, for noetherian integral local rings of dimension one the statements "*integrally closed*" and "*regular*" are equivalent. Therefore, we use *normalization* for resolution of singularities of X. We now describe this here:

Let $\{(U_i)\}$ be an open affine cover of X, for each open subset U_i of X let $A_i = \mathcal{O}_X(U_i)$ and A_i' be the integral closure of A_i in $\mathrm{Frac}(A_i)$. Then the injection $A_i \to A_i'$ induces a morphism π_i: $U_i' \to U_i$ for each i where U_i' =SpecA_i'. The π_i are called normalization morphisms. Then gluing the normalizations π_i: $U_i' \to U_i$ we get a finite surjective morphism $\pi : X' \to X$ which is the normalization of X.

Lemma 1.1.4. *Let X and X' be the same as above. There is an exact sequence of coherent sheaves on X,*

$$0 \to \mathcal{O}_X \to \pi_*\mathcal{O}_{X'} \to \mathcal{S} \to 0 \tag{1.3}$$

such that $\mathcal{S}$ is a skyscraper sheaf.

Proof. The normalization morphism $\pi : X' \to X$ of schemes is endowed with a sheaf morphism $f : \mathcal{O}_X \to \pi_*\mathcal{O}_{X'}$. Let U=Spec$A$ be an open affine subset of X and A'= Spec$(\mathcal{O}_{X'}(\pi^{-1}(U)))$ be the integral closure of A in Frac(A). The restriction of f on U is exactly the injection map i: $A \to A'$ which means f is locally injective hence f is injective globally. Let $\mathcal{S}$ be the sheaf associated to presheaf coker$f$ which makes the sequence (1.3) exact.

Let V=Spec$B$ be an open affine subset of X not containing a singular point of

X. Then the integral closure of B in Frac(B) is again B so the restriction morphism f on V is surjective. This implies that the support of the sheaf $\mathcal{S}$ contains only the singular points of X which is finite. Hence, $\mathcal{S}$ is a skyscraper sheaf on X and $\mathcal{S}(U)=\bigoplus_{P\in U}\left(\frac{\mathcal{O}'_{X,P}}{\mathcal{O}_{X,P}}\right) = \bigoplus_{P\in U}\mathcal{S}_P$ where $\mathcal{O}'_{X,P}$ is the integral closure of $\mathcal{O}_{X,P}$ in Frac$(\mathcal{O}_{X,P})$. It is clear that the stalk $\mathcal{S}_P = 0$ if P is not a singular point. □

Remark 1.1.5. Let P be a singular point of X, the integral closure of $\mathcal{O}_{X,P}$ in its fraction field is $\bigcap_{i=0}^{m}\mathcal{O}_{X',Q_i}$ where $\pi^{-1}(P) = \{Q_1, \ldots, Q_m\}$.

Definition 1.1.6. Let P be the same as above and $\pi^{-1}(P) = \{Q_1, \ldots, Q_m\}$ where $\pi : X' \to X$ is the normalization morphism. We say that P is an *ordinary singular point* if for an open affine neighborhood U of P

$$\mathcal{O}_X(U) = \{f \in \mathcal{O}_{X'}(\pi^{-1}(U)) | f(Q_1) = \cdots = f(Q_m)\} \tag{1.4}$$

Lemma 1.1.7. *Let X be a singular curve with only ordinary singular points and $\pi : X' \to X$ is the normalization morphism. Then there exists a surjective homomorphism $\pi^* : Pic(X) \to Pic(X')$ such that the kernel of it consists of a number of copies of $\mathbb{G}_m(k)$.*

Proof. We just sketch the basic idea of the proof and for detailed proof, see [16, Lemma 7.5.12]. Let P be an ordinary singular point of X such that U=Spec$A$ is an open affine neighborhood of P , $\pi^{-1}(P) = \{Q_1, \ldots, Q_n\}$ and $B = \mathcal{O}'_X(\pi^{-1}(U))$. By lemma 1.1.4, we have a short exact sequence

$$0 \to \mathcal{O}_X \to \pi_*\mathcal{O}'_X \to \mathcal{S} \to 0.$$

This complex locally around P is

$$0 \to A \to B \to \frac{B}{A} \to 0.$$

Then the homomorphisms $B^* \to (k^*)^n$, $f \mapsto (f(Q_1), \ldots, f(Q_n))$ and $A^* \to k^*$, $a \mapsto a(P)$ induce an isomorphism:

$$\rho : \frac{B^*}{A^*} \to \frac{(k^*)^n}{\Delta(k^*)} \simeq (k^*)^{n-1}$$

where $\Delta : k^* \mapsto (k^*)^n$ is the diagonal homomorphism. □

Let $P \in X$ be a singular point such that $\pi^{-1}(P) = \{Q_1, \ldots, Q_n\}$ and U be an open neighborhood of P in X. Then define the open affine curve:

$$V\text{=Spec}\{f \in \mathcal{O}_{X'}(\pi^{-1}(U)) | f(Q_1) = \cdots = f(Q_n)\}$$

As the singular point P varies in X, gluing the corresponding open affine curves V, we get a curve Y in between X and X' such that the normalization morphism $\pi : X' \to X$ factors through $\pi_1 : X' \to Y$ and π_2: Y→ X. We are going to use the morphism π_2 to get the structure of the Picard Group of X.

Lemma 1.1.8. *By using the same notations as above, the morphism $\pi_2 : Y \to X$ induces a surjective homomorphism π_2^*: Pic(X) →Pic(Y) whose kernel is unipotent with dimension $g(X) - g(Y)$ where $g(X)$ is the arithmetic genus of X and $g(Y)$ is the arithmetic genus of Y.*

Proof. By lemma 1.1.4 we have an exact sequence

$$0 \to \mathcal{O}_X^* \to \pi_{2*}O_Y^* \to \bigoplus_P \mathcal{S}_P^* \to 0$$

where $\mathcal{S}_P = \left(\dfrac{\mathcal{O}^*_{Y,P}}{\mathcal{O}^*_{X,P}} \right)$, which induces an exact cohomology sequence

$$0 \to \mathcal{O}_X(X)^* \to \mathcal{O}_Y(Y)^* \to \mathcal{S} \to Pic(X) \to Pic(Y) \to 0 \tag{1.5}$$

Hence π_2^*: Pic(X) →Pic(Y) is onto with kernel $\mathcal{S}$ and from [16, Lemma 7.5.12 and Lemma 7.5.18] $\mathcal{S}$ is a unipotent group of dimension $g(X) - g(Y)$. □

1.2 Polynomial Factorization Modulo p

In this section, we briefly describe the well-known algorithms, Cantor-Zassenhaus (C-Z) [7] and Berlekamp [4], for polynomial factorization modulo a prime number p. We assume the field $k = \mathbb{Z}/p\mathbb{Z}$ through the section. We also note that there is no deterministic polynomial time algorithm to find the factors of polynomials in k. The general strategy for polynomial factorization modulo p can be summarized as follows:

Let $F(x)$ be a monic polynomial in $k[x]$.

Step 1: *Square Free Factorization:* Find square-free polynomials $F_i(x)$, $i = 1, \ldots, n$ such that

$$F(x) = F_1(x) \cdot F_2(x)^2 \cdots F_n(x)^n \tag{1.6}$$

and $F_i(x)$ are coprime.

Step 2: *Distinct Degree Factorization:* For each $F_i(x)$, find $F_{i,d}(x)$ such that $F_i = \prod F_{i,d}(x)$ and $F_{i,d}$ is the product of irreducible factors of $F_i(x)$ with degree d.

Step 3: *Final Splitting:* Find irreducible factors of each $F_{i,d}(x)$.

The important step is the step of Final Splitting and the well-known methods for this step are all probabilistic. We now give a brief description of an algorithm for each step. More details can be found in [9, Chapter 3].

1.2.1 Square Free Factorization (SFF)

Although there are many algorithms for SFF, all of them are the variations of the following method.

Let $F = F_1 \cdot F_2^2 \cdots F_n^n$ be a polynomial such that the F_i are square-free and coprime. Then

$F' = \sum_{0<i<n+1} F_1 \cdots F_{i-1}^{i-1} \cdot i \cdot F_i^{i-1} \cdot F_i' \cdot F_{i+1}^{i+1} \cdots F_n^n$

1. Compute $D =$gcd(F, F') which yields $D = F_2 \cdot F_3^2 \cdots F_n^{n-1}$
2. Compute $V = F/D$ which yields $V = F_1 \cdot F_2 \cdots F_n$
3. Compute $T =$gcd (D, D') which yields $T = F_3 \cdot F_4^2 \cdots F_n^{n-2}$
4. Compute $T_0 = D/T$ which yields $T_0 = F_2 \cdot F_3 \cdots F_n$
5. Compute V/D_0 which yields F_1
6. Assign $F = D$ and go to (1)

This is one of the necessary steps for C-Z and Berlekamp's algorithms as well as our algorithm that we present in chapter 4. The second step of polynomial factorization is Distinct Degree Factorization. This step is necessary only for the C-Z algorithm. Once we finish SFF step, Berlekamp and our algorithm can be used for final splitting.

1.2.2 Distinct Degree Factorization

Let $F_i(x)$ be a square-free factor of $F(x)$ and $F_{i,d}$ be a factor of $F_i(x)$ such that $F_{i,d}$ is the product of irreducible factors of $F_i(x)$ of degree d. Let $T(x)$ be an irreducible polynomial of degree d. Then $K = k[x]/T(x)$ is a finite field with p^d elements. Hence any non-zero element of K satisfies the equation $x^{p^d} - x$. This means that $T(x)$ is a factor of $x^{p^d} - x$ in $k[x]$. On the other hand each irreducible factor of $x^{p^d} - x$ which is not a factor of $x^{p^c} - x$ for $c < d$ has exactly degree d. By using this idea we can easily find $F_{i,d}$ by using the following method:

1. $T_1(x) = F_i(x)$

2. $B_{r+1}(x) = F_i(x)/\text{gcd}(T_r(x), x^{p^r} - x)$ for $r = 2, \ldots, d$

3. $F_{i,d}(x) = gcd(F_i(x), B_d(x))$

1.2.3 Final Splitting

The C-Z algorithm can do Final Splitting in an efficient way, but in many cases especially for a small prime number p, Berlekamp's algorithm is much better than the C-Z algorithm. As we mentioned above Berlekamp's Algorithm can be used right after Square Free Factorization.

1.2.3.1 Cantor-Zassenhaus Split

This algorithm is based on the following observation:

Theorem 1.2.1. *Let $F_{i,d}(x)$ be the same as above. Then for any polynomial $G(x) \in k[x]$ we have*

$$F_{i,d}(x) = gcd(F_{i,d}(x), G(x)) \cdot gcd(F_{i,d}(x), G(x)^{(p^d-1)/2}+1) \cdot gcd((F_{i,d}(x), G(x)^{(p^d-1)/2}-1) \tag{1.7}$$

Proof. See [9, Proposition 3.4.5] □

Cantor-Zassenhaus Algorithm: Let $F_{i,d}(x)$ be the same as above and p be an odd prime. This algorithm finds the irreducible factors of $F_{i,d}(x)$.

1. Randomly select a monic polynomial $G(x) \in k[x]$ of degree less than $2d$.

2. Set $H(x) = \gcd(F_{i,d}(x),\ G(x)^{(p^d-1)/2} - 1)$. If $H(x) = 1$ or $H(x) = F_{i,d}(x)$ go to (1) otherwise go to (3)

3. Find the factors of $H(x)$ and $F_{i,d}(x)/H(x)$ by using this algorithm

The probability that C-Z algorithm gives a non-trivial factor of $F_{i,d}(x)$ in a single trial is closed to 1/2 [9]. However, for Berlekamp's algorithm, which we present now, this probability is always less than 1/2.

1.2.3.2 Berlekamp's Algorithm

Let $F_i(x)$ be a reducible square-free polynomial in $k[x]$. This algorithm first finds polynomials $G(x)$ such that

$$G(x)^p \equiv G(x) \quad (mod \quad F_i(x)) \tag{1.8}$$

These polynomials form a subalgebra which is called the Berlekamp subalgebra. The polynomials $G(x)$ can be found after constructing a basis for Berlekamp subalgebra. Then for each $G(x)$ recursively compute $\gcd(G(x)-s, F_i(x))$ for each $s \in k$ until the result is a non-trivial factor of $F_i(x)$. The running time of Berlekamp's algorithm depends on p, but it is arguably the most accepted one in practice and it is being used in some computer software like PARI/GP.

1.3 Computing Square Roots in Finite Fields

An important problem in computational number theory is the computation of square roots mod p. Although there are some deterministic algorithms working in some cases, Shanks-Tonelli's [22] probabilistic algorithm is the most widely accepted one in practice. There is also a deterministic algorithm for this problem by R. Schoof [20] but it can only be used for computing square roots of small size numbers, since its running time depends on the size of a number of which one wants to compute square root. We use Schoof's algorithm in one of our algorithms to compute a square root of 3 modulo p. In this section we present a brief summary of these algorithms.

More details can be found in [9]. Our algorithms for this problem will be introduced in chapter 3.

1.3.1 Shanks' Algorithm

Let p be an odd prime number and $\mathbb{F}_p$ be a finite field with p elements. For a given number a , we want to find $x \in \mathbb{F}_p$ such that $x^2 = a \pmod{p}$. Suppose we know such an x exists in $\mathbb{F}_p$, i.e. $\left(\frac{a}{p}\right)$=1. There is an easy method for some primes to find x. For example for primes $p \equiv 3 \pmod{4}$ we can say that $x \equiv a^{(p+1)/4} \pmod{p}$ is a square root of a. Since

$$x^2 \equiv a^{(p+1)/2} \equiv a^{(p-1)/2}a \equiv 1 \cdot a (\bmod\ p)$$

because $\left(\frac{a}{p}\right) = 1 \equiv a^{(p-1)/2} \pmod{p}$.

For the half of the remaining primes, that is for $p \equiv 5 \pmod{8}$, there is also a trivial method to find a square root of a in $\mathbb{F}_p$. Because if we have $p \equiv 5 \pmod{8}$ and $a^{(p-1)/2} \equiv 1 \bmod (p)$ then $a^{(p-1)/4} \equiv \pm 1 \pmod{p}$.

If $a^{(p-1)/4} \equiv -1 \pmod{p}$ consider $x \equiv 2a(2^2a)^{(p-5)/8} \pmod{p}$. Then

$$x^2 \equiv 2^2a^2(2^2)^{(p-5)/4}a^{(p-5)/4} \equiv 2^{(p-1)/2}aa^{(p-1)/4} \bmod (p).$$

Since $a^{(p-1)/4} \equiv -1 (\bmod\ p)$ and $\left(\frac{2}{p}\right) = (-1)^{(p^2-1)/8} = -1$, we have $x^2 \equiv a \pmod{p}$.

For the other case i.e. $a^{(p-1)/4} \equiv 1 \pmod{p}$, similarly we can show that if $x \equiv -2a(2^2a)^{(p-5)/8} \pmod{p}$ then $x^2 \equiv a \pmod{p}$.

For primes $p \equiv 1 \pmod{8}$, there is no deterministic polynomial time algorithm to compute square roots mod p. Shanks' algorithm is the only one used in practice for these primes. We now describe this algorithm here.

The cyclic multiplicative group $\mathbb{F}_p^*$ is of order $p-1$ which is an even number. Suppose H is the 2-Sylow subgroup of $\mathbb{F}_p^*$ and z is a generator of H. Then if $p-1 = 2^e k$ where k is odd , the order of z is 2^e. Let $b = a^k$. Since $a^{(p-1)/2} \equiv (a^k)^{2^{e-1}} \equiv 1 \pmod{p}$, we conclude $b \in H$. Hence $b \equiv z^r (\text{mod } p)$ for a number r. Note that r is even, since $b^{2^{e-1}} \equiv 1 = z^{r2^{e-1}} \pmod{p}$ and this implies 2^e divides $r2^{e-1}$. Similarly $b^{-1} \equiv z^{2^e - r} \equiv z^t \pmod{p}$ so t is also even. Then

$$x \equiv a^{(k+1)/2} z^{t/2} \pmod{p} \text{ is a square root of } a \text{ since}$$

$$x^2 \equiv a^{k+1} z^t \equiv a^k z^t . a \equiv b b^{-1} a \equiv a (\text{mod } p).$$

Now the important problem is how to find a generator z for the 2-Sylow subgroup H of $\mathbb{F}_p^*$. One can show that for any v such that $\left(\dfrac{v}{p}\right) = -1$, $z = v^k$ is a generator of H. Although there is no explicit way to find such a z for all primes , for a random $d \in \mathbb{F}_p$ there is 1/2 chance that $\left(\dfrac{d}{p}\right) = -1$. This is the only probabilistic part of this algorithm and the hardest part is to find t such that $b^{-1} \equiv z^t$ (mod p). For detailed analysis of this algorithm see [9, Chapter 1]. The expected running time of this algorithm $O(ln^4 p)$.[9, Section 1.5].

1.3.2 Schoof's Algorithm

Schoof's Algorithm for the equation $x^2 - a \equiv 0 \pmod{p}$ gives a solution deterministically but the running time of this algorithm depends on the size of a.

This makes Schoof's Algorithm non-practical in practice for large numbers. We now give a brief summary of this method.

Assume $a \in \mathbb{F}_p$ is a square. We may assume $a < 0$ since $a \equiv a - p \pmod{p}$. We can find an elliptic curve E with complex multiplication by $\mathbb{Q}(\sqrt{a})$ and reduce it mod p.[23] Actually it is not easy in practice to find such a curve for a except for very small size a. This is the only part that makes running time this method depend on $|a|$.

Now suppose we get an elliptic curve E with complex multiplication by $\mathbb{Q}(\sqrt{a})$ and reduce it mod p. Let

$$d = p + 1 - \#E(\mathbb{F}_p) = p + 1 - deg(\phi_p - 1)$$

where $\#E(\mathbb{F}_p)$ is the number of points on E over the field $\mathbb{F}_p$ and ϕ_p is the Frobenious morphism. Consider the polynomial $h(x) = x^2 - dx + p$ and assume that α is a root of $h(x)$. Then we have

$$\alpha\overline{\alpha} = p$$

$$\alpha + \overline{\alpha} = d$$

Since the elliptic curve E has complex multiplication by $\mathbb{Q}(\sqrt{a})$, α and $\overline{\alpha} \in \mathbb{Q}(\sqrt{a})$, i.e. $\alpha = u + \sqrt{a}v$ where $2u$, $2v \in \mathbb{Z}$. Then

$$d = \alpha + \overline{\alpha} = (u + \sqrt{a}v) + (u - \sqrt{a}v) = 2u \text{ and}$$

$$p = \alpha\overline{\alpha} = u^2 - av^2$$

We get d by computing $\#E(\mathbb{F}_p)$. We can also compute an integer square root of $\dfrac{u^2 - p}{a}$ in a relatively fast way.[9, Section 1.7] Since we have $p = u^2 - av^2$ then

$\left(\frac{u}{v}\right)^2 \equiv a \pmod{p}$.

We use Schoof's Algorithm in one of our algorithms to compute square root of -3 and -1. One can show that the elliptic curves $E_1 : y^2 = x^3 - x$ and $E_2 : y^2 = x^3 - 1$ have complex multiplication by $\mathbb{Q}(\sqrt{-1})$ and $\mathbb{Q}(\sqrt{-3})$ respectively. Hence for computing a square root of -1 or -3 mod p, we just need to know $\#E_1(\mathbb{F}_p)$ and $\#E_2(\mathbb{F}_p)$.

Chapter 2

Computing in the Jacobian of a Singular Hyperelliptic Curve

2.1 Introduction

Cantor's Algorithm[CAN] gives an efficient way for computing in the Jacobian of a smooth hyperelliptic curve. This algorithm relies on the Mumford Representation[17] of the points in Jacobians. This compact representation of points in Jacobians and Cantor's algorithm make non-singular hyperelliptic curves suitable for many applications in cryptography. In this chapter we show the extension of the Mumford representation for singular hyperelliptic curves. The algorithms that we present in the following chapters mainly depend on this extension. We also investigate the behavior of Cantor's algorithm for singular hyperelliptic curves.

The use of non-singular hyperelliptic curves, especially lower genus ones, mainly depends on the hardness of the Discrete Logarithm Problem (DLP) on their Jacobians in a finite field. The extension of Cantor's Algorithm for singular hyperelliptic curves raises the natural question: is DLP hard in the Jacobian of a singular hyperelliptic curve in a finite field? The question has an answer for genus 1. For higher genus the answer is almost the same,[15] i.e. the DLP in the Jacobian of a singular hyperelliptic curve is at most hard as the DLP in the multiplicative group of the finite field.

We first describe the Mumford representation and Cantor's algorithm for non-singular hyperelliptic curves.

2.2 Smooth Hyperelliptic Curves

Let k be an algebraically closed field with characteristic different from 2. A complete irreducible curve X is called hyperelliptic if there is a morphism $h : X \to \mathbb{P}^1$ of degree 2. For our purposes we assume a non-singular affine hyperelliptic curve is defined by an algebraic equation $y^2 = f(x)$ where $f(x)$ is of degree $2g + 1$ without repeated roots in k. The coordinate ring $k[X]$ of X is a Dedekind domain and it is the algebraic closure of the polynomial ring $k[x]$ in the function field $k(X)$ of X. This observation leads to a connection between the Jacobian of X and the ideal class group of an imaginary quadratic field. Actually, Cantor's algorithm is assumed to be an analogue of the method of composition of binary quadratic forms in the ideal class group of an imaginary quadratic field.

Consider the plane the curve $Y \subset \mathbb{P}^2_k$ defined by the homogeneous form of $y^2 = f(x)$. Then the curve X is the partial normalization of Y at the point at infinity, since there is a singularity at the point at infinity of Y. Note that the geometric genus of X is denoted by g.

The Jacobian, $\mathrm{Jac}(X)$, of X is defined as the group of (Weil) divisors of degree zero modulo principal divisors. More details and proof of the statements here can be found in [24, Chapter 13].

2.2.1 The Mumford Representation for Smooth Hyperelliptic Curves

Let $P_i = (x_i, y_i)$ be a point on X. The map $\omega : X \to X$, $\omega(x_i, y_i) = (x_i, -y_i)$ is called the **hyperelliptic involution**. Let D be a divisor class in Jac(X). Then D can be written as $\sum\limits_i n_i([P_i] - [\infty])$.

Definition 2.2.1. Let $D = \sum\limits_i n_i([P_i] - [\infty])$ with $P_i = (x_i, y_i)$ be a divisor class in Jac(X). D is called a **reduced divisor** if it satisfies the following:

1. $n_i \geq 0$ for all i
2. if $y_i = 0$ then $n_i = 0$ or 1
3. if $[P_i]$ with $y_i \neq 0$ occurs in the sum, then $[\omega(P_i)]$ does not occur.
4. $\sum\limits_i n_i \leq g$.

There is a unique reduced divisor D for each divisor class in Jac(X) [24, Proposition 13.6]. The representation $\sum\limits_i n_i([P_i] - [\infty])$ of a divisor class is not concrete enough to perform group operations efficiently in Jac(X). The following representation, which is called the Mumford Representation of a reduced divisor, is a more suitable representation for computational applications.

Theorem 2.2.2. *Let $D = \sum\limits_i n_i([P_i] - [\infty])$ be a divisor class in Jac(X) with $n_i \geq 0$. Then there exits a pair $(U(x), V(x))$ of polynomials corresponding to D satisfying the following.*

1. U(x) is monic

2. $deg(V(x)) < deg(U(x)) = \sum_i n_i$

3. $V(x)^2 - f(x)$ *is divisible by* $U(x)$.

Proof. See [24, Theorem 13.7]. □

2.2.2 Cantor's Algorithm for Smooth Hyperelliptic Curves

We can perform the group operation on the Jacobian of a non-singular hyperelliptic curve by using only polynomial arithmetic over the field k with the help of above representation of divisor classes. This method is due to David Cantor. [6]

Cantor's Algorithm: This algorithm takes two divisor classes $D_1 = [u_1(x), v_1(x)]$ and $D_2 = [u_2(x), v_2(x)]$ on X and outputs the unique reduced divisor D such that $D = D_1 + D_2$.

```
1. $d = gcd(u_1, u_2, v_1 + v_2)$ with polynomials $h_1, h_2, h_3$ such that
   $d = h_1u_1 + h_2u_2 + h_3(v_1 + v_2)$
2. $u = \frac{u_1u_2}{d^2}$ and $v \equiv \frac{h_1u_1v_2 + h_2u_2v_1 + h_3(v_1v_2 + f)}{d} \pmod{u}$
   repeat:
3. $\widetilde{u} = \frac{v^2 - f}{u}$ and $\widetilde{v} \equiv v \pmod{\widetilde{u}}$
4. $u = \widetilde{u}$ and $v = \widetilde{v}$
   until deg $(u) \leq g$
5. multiply $u$ by a constant to make $u$ monic.
```

Note that the Mumford Representation[17] and Cantor's Algorithm[6] works for any field of odd characteristic and genus.

2.3 Computing in Jacobians of Singular Hyperelliptic Curves

We show in this section that the above methods for smooth hyperelliptic curves can be extended to singular hyperelliptic curves. The curve $X : y^2 = f(x)$ is singular if $f(x)$ has a multiple root (deg$f(x)$ is still $2g+1$). The curve X' is the normalization of X. The singular points of X are of the form $(a, 0)$ where a is a root of $f(x)$ with multiplicity greater than 1. Note that the arithmetic genus of the curve is g. In the remaining part of this section we prove the following statement: any divisor class D in Jac(X) has a unique representative $(u(x), v(x))$ satisfying the following:

1. $u(x)$ is a monic polynomial in $k[x]$

2. deg$(v(x))$ <deg$(u(x)) \leq g$

3. $v(x)^2 - f(x)$ is divisible by $u(x)$.

4. if $u(x)$ and $v(x)$ are multiples of $(x - a)$ for a singular point $(a, 0)$ then $\dfrac{f(x) - v(x)^2}{u(x)}$ is not a multiple of $(x - a)$.

We also show that Cantor's Algorithm works as the same way as in the non-singular case.

2.3.1 The Mumford Representation for Singular Hyperelliptic Curves

Definition 2.3.1. Let X be a singular hyperelliptic curve defined by an algebraic equation $y^2 = f(x)$ where $f(x)$ is a polynomial of degree $2g + 1$. Let S be the set of singular points of X and $k[X]$ be the coordinate ring of X and $k(X)$ be its function field. We define

1. Pic(X)={ group of 1-cycles $D = \sum_i n_i P_i, \quad P_i \in X - S$ modulo: $D \backsim 0$ if $D = div(h(x))$ for some $h(x)$ in the function field $k(X)$ of X, $h(x)$ is continuous, finite and nonzero at each point in S. }

2. Pic(X) $\cong CaCl(X)$ = the group of isomorphism classes of Cartier divisors modulo linear equivalence.

3. Pic(X) is the group of all isomorphism classes of invertible (locally free of rank 1) $\mathcal{O}_X$-modules.

Remark 2.3.2. The three definitions of Pic(X) are equivalent. ([16, Proposition 7.1.18 and Corollary 7.1.19], [17, Section 3], [21, Chapter 11])

Definition 2.3.3. Jac(X) is the identity component Pic^o(X) of the algebraic group Pic(X) [5, Chapter 8] which is the degree zero divisor classes of Pic(X).

Theorem 2.3.4. *Let D be in Jac(X). Then D corresponds to a pair of polynomials* $(U(x), V(x))$ *satisfying*

1. $V(x)^2 - f = U(x)\widetilde{U}(x)$ *for some polynomial* $\widetilde{U}(x)$*.*

2. $U(x)$ *is monic.*

3. *deg*$(V(x))$ <*deg*$(U(x))$*.*

4. $U(x)$ *is not zero at singular points of* X*.*

Remark 2.3.5. There is one point at infinity which is (0:1:0) and denoted by ∞. Although this point is singular for the curve $X : y^2 = f(x)$ if deg$(f(x)) \geq 2$, we always assume that X is the partial normalization of the curve defined by the homogeneous form of the curve $y^2 = f(x)$ at the point infinity.

Proof. Let D be in $\mathrm{Pic}^o(X)$. Then $D = \sum_i n_i(Q_i)$ where $Q_i = (q_i, r_i)$ are non-singular points of X. The divisor of the form $(Q_j) + \omega(Q_j) - 2(\infty)$ is a divisor of a function $x - q_j$ where again $Q_i = (q_j, r_j)$ are non-singular points of X. Now by adding suitable multiples of the divisors $(Q_i) + \omega(Q_i) - 2(\infty)$ to D, we may assume $D = \sum_i n_i((Q_i) - (\infty))$ where $n_i > 0$, Q_i and $\omega(Q_i)$ are not in the sum simultaneously. Then consider the function $U(x) = \prod_i (x - q_i)^{n_i}$. We may solve the congruences $W(x)^2 = f(x) \pmod{(x - q_i)^{n_i}}$ for each i as in [24, Theorem 13.5]. Then combining all solutions by using the Chinese Remainder Theorem we get a polynomial $\widetilde{V}(x)$ such that $V(x) \equiv \widetilde{V}(x) \pmod{U(x)}$ is the polynomial satisfying above properties.

Let $D_1 = \sum_j n_j((Q_j) - (\infty))$ and $D_2 = \sum_j m_j((Q_j) - (\infty))$ be two divisors in $\mathrm{Jac}(X)$ where Q_i, Q_j are non-singular points of X. Define

$$\gcd(D_1, D_2) = \textstyle\sum_j \min\{n_j, m_j\}((Q_j) - (\infty)).$$

Consider the polynomial $y - V(x)$ which is a principal Cartier divisor of X. By [24, Proposition 13.4] $\mathrm{div}(y - V(x))$=$\sum_i m_i((Q_i) - (\infty))$+$\sum_j m_j((P_j) - (\infty)))$ where $m_i \geq n_i$ and P_i's are different from Q_j's. Then we see that

$D = \gcd(\mathrm{div}(U(x)), \mathrm{div}(y - V(x)))$. □

Definition 2.3.6. Let X be a hyperelliptic curve defined by the algebraic equation $y^2 - f(x)$ where $f(x)$ is a monic of odd degree polynomial. Let $A = \mathcal{O}_X(X)$ and $\mathcal{H}_X$ be the set of pairs of polynomials $(U(x), V(x))$ in A satisfying

1. $U(x)$ is a monic polynomial.

2. $V(x)^2 - f(x)$ is divisible by $U(x)$.

3. $\deg(U(x)) > \deg(V(x))$

The pair $(U(x), V(x))$ in $\mathcal{H}_X$ is called a **singular** if for some singular point $(a, 0)$ of X, $(x-a)$ divides $U(x)$, $V(x)$, and $\frac{V(x)^2 - f}{U(x)}$.

2.3.2 The Reduction Algorithm

The reduction algorithm is the key operation to get a unique representative of a divisor class in a Jacobian. Let X be the same as above and $(U(x), V(x))$ be a pair of polynomials in $\mathcal{H}_X$ representing the divisor D in $\text{Jac}(X)$. We do :

1. $\widetilde{U}(x) = \frac{V(x)^2 - f(x)}{U(x)}$

2. $\widetilde{V}(x) \equiv -V(x) \pmod{\widetilde{U}(x)}$

3. set $U(x) = \widetilde{U}(x)$, $V(x) = \widetilde{V}(x)$

4. Multiply $U(x)$ by a constant to make $U(x)$ monic.

Proposition 2.3.7. *Let $(U(x), V(x))$ be a non-singular pair in $\mathcal{H}_X$ and X be a singular hyperelliptic curve defined by $y^2 = f(x)$. Suppose the reduction algorithm on $(U(x), V(x))$ produces $(\widetilde{U}(x), \widetilde{V}(x))$. Then the new pair $(\widetilde{U}(x), \widetilde{V}(x))$ is also a non-singular element of $\mathcal{H}_X$.*

Proof. Suppose $(a, 0)$ is a singular point of X. If $(x-a)$ divides $U(x)$, then $(x-a)$ divides $V(x)$ since $(x-a)$ divides $f(x)$ and $f(x)-V(x)^2$ is divisible by $U(x)$. Then $(x-a)$ does not divide $\widetilde{U}(x) = \dfrac{V(x)^2 - f(x)}{U(x)}$, since $(U(x), V(x))$ is non-singular. Hence the new the pair $(\widetilde{U}(x), \widetilde{V}(x))$ is non-singular.

Now suppose $(x-a)$ does not divide $U(x)$ but divides $\widetilde{U}(x) = \dfrac{V(x)^2 - f(x)}{U(x)}$. That means $(x-a)$ is a factor of $V(x)$. We know $\widetilde{V}(x) \equiv -V(x) \pmod{\widetilde{U}(x)}$, i.e. $\widetilde{V}(x) = -V(x) + h(x)\widetilde{U}(x)$ for some polynomial $h(x)$, so $(x-a)$ is also a factor of $\widetilde{V}(x)$. Now $(x-a)$ divides both members of $(\widetilde{U}(x), \widetilde{V}(x))$ but

$$\widetilde{\widetilde{U}}(x) = \frac{\widetilde{V}(x)^2 - f(x)}{\widetilde{U}(x)} = \frac{(-V(x) + h(x)\widetilde{U}(x))^2 - f(x)}{\widetilde{U}(x)}$$

$$= \frac{V(x)^2 - 2V(x)h(x)\widetilde{U}(x) + h(x)^2\widetilde{U}(x)^2 - f(x)}{\widetilde{U}(x)}$$

$$= \frac{V(x)^2 - f(x)}{\widetilde{U}(x)} + \frac{-2V(x)h(x)\widetilde{U}(x) + h(x)^2\widetilde{U}(x)^2}{\widetilde{U}(x)}$$

$$= U(x) - 2V(x)h(x) + h(x)^2\widetilde{U}(x)$$

is not divisible by $(x-a)$. Hence the new pair $(\widetilde{U}(x), \widetilde{V}(x))$ is also non-singular. □

Lemma 2.3.8. *Let D be a divisor in Jac(X) corresponding to the non-singular pair $(U(x), V(x))$ in $\mathcal{H}_X$. Suppose that by applying the reduction algorithm we get the pair $(\widetilde{U}(x), \widetilde{V}(x))$. Then $(\widetilde{U}(x), \widetilde{V}(x))$ also represents the same divisor class as $(U(x), V(x))$.*

Proof. We are using definition 1 for $\mathrm{Pic}(X)$, i.e. $D = \sum_i n_i P_i$, and P_i's are non-singular points of X. By construction, $U(x)$ does not have a factor $(x - a_i)$ for any singular point $(a_i, 0)$ of X. Hence the pair $(U(x), V(x))$ is non-singular.

Let $A = \mathcal{O}_X(X) = k[x,y]/(y^2 - f)$. We have $y^2 - f = y^2 - V^2 + U\widetilde{U} = (y+V)(y-V) + U\widetilde{U} = 0$ in A. Then the ideal $I = (U(x), y - V(X)) \supset (y^2 - f(x))$ has the same support with D and it defines the zero dimensional subscheme D of X.

First we show that any ideal $I = (U(x), y - V(x))$ of A such that $(U(x), V(x))$ is non-singular in $\mathcal{H}_X$ is an A-module of rank 1. For this, it is enough to show $I_m \cong A_m$ for all maximal ideals of A. The isomorphism is clear when $I \not\subseteq m$. Assume $I \subseteq m$ where $m = (x - a, y - b)$ for some $a, b \in \mathbb{A}^1_k$ and $b \neq 0$. We have $I \subseteq m$ and this implies $b = V(a)$, hence $y + V(x) \notin m$. Then $y - V(x) = \dfrac{U(x)\widetilde{U}(x)}{y + V(x)}$ which means $I_m = (U(x))_m \cong A_m$. Now suppose $b = 0$. Then $(x - a)$ is a factor of $U(x)$ and $V(x)$. Since the pair $(U(x), V(x))$ is non-singular, $(x - a)$ does not divide $\widetilde{U}(x)$. Then the equality $U(x) = \dfrac{(y - V(x)(y + V(x)}{\widetilde{U}(x)}$ implies that $I_m = (y - V(x))_m \cong A_m$. Therefore the ideal $(U(x), y - V(x))$ is an invertible ideal of A.

Now consider the ideal $J = (\widetilde{U}(x), y + V(x))$. The ideal J corresponds to a non-singular pair $(\widetilde{U}(x), -V(x))$ of $\mathcal{H}_X$. Therefore it is also an A-module of rank 1. We have $J = (\widetilde{U}(x), y + V(x)) = (\widetilde{U}(x), y - (-V(x)) = (\widetilde{U}(x), y - (-V(x)\mathrm{mod}(\widetilde{U}(x))) = (\widetilde{U}(x), y - \widetilde{V}(x))$. Note that $J = \left(\dfrac{y + V(x)}{U(x)}\right) I$. Hence I and J are isomorphic invertible A-submodules of $\mathrm{Frac}(A)$ which means they represent the same class in Pic(X)=CaCl(X). $\square$

Definition 2.3.9. Let $(U(x), V(x))$ be a non-singular element of $\mathcal{H}_X$ and g be the arithmetic genus of X. We call the pair $(U(x), V(x))$ **reduced** if deg$U(x) \leq g$.

Remark 2.3.10. Note that the arithmetic genus of X defined by $y^2 = f(x)$ where deg$f(x) = 2g + 1$ is g and its geometric genus is the arithmetic genus of the normalization X' of X.

Theorem 2.3.11. *Let $D = \sum_i n_i Q_i$ be a divisor class in Jac(X) such that Q_i are non-singular points. Then there exists a unique reduced pair $(\widetilde{U}(x), \widetilde{V}(x))$ representing the class of D.*

Proof. Let $(U(x), V(x))$ be a non-singular representative of D such that deg$(U(x))$ is greater than the arithmetic genus g of X. Then we have deg$f(x)$ <2deg$U(x)$ and deg$V(x)^2 <$ 2deg$U(x)$. Since $U(x)\widetilde{U}(x) = f(x) - V(x)^2$ and deg$(f(x) - V(x)^2) <$ 2deg$U(x)$, we get deg$\widetilde{U}(x)$ <deg$U(x)$. This means that in each step of the reduction algorithm the degree of the pair is decreasing. Therefore applying the reduction algorithm finitely many times to $(U(x), V(x))$, we get a reduced non-singular pair $(\widetilde{U}(x), \widetilde{V}(x))$ in the same divisor class as D. As for uniqueness we need to use the definitive form of the Riemann-Roch Theorem. Since we already know from chapter 1 that the curve X is a Gorenstein curve, and the Riemann-Roch is similar to the case where X is non-singular except that we use arithmetic genus instead of geometric genus. Actually for smooth curves these two numbers are equal. For the rest of the proof see [24, Proposition 13.6]. □

Cantor's Algorithm Let $D_1 = \sum_i n_i P_i$ and $D_2 = \sum_j m_j Q_j$ be divisors in $Pic^o(X)$ corresponding to pairs (U_1, V_1) and (U_2, V_2) with P_i, Q_j non-singular points.

1. Let d=gcd $(U_1, U_2, V_1 + V_2)$ with polynomials h_1, h_2, h_3 such that $d = U_1h_1 + U_2h_2 + (V_1 + V_2)h_3$.

2. Let $V_0 = (U_1V_2h_1 + U_2V_1h_2 + (V_1V_2 + f)h_3/d$.

3. Let $U = U_1U_2/d^2$ and $V \equiv V_0(\text{mod } U)$ with deg$V <$ deg U.

4. Multiply U by a constant to make it monic.

The pair (U, V) is the Mumford representation of the divisor class of $D_1 + D_2$.

Proof. See [24, Theorem 13.10]. The resulting pair (U, V) may not be reduced but it is non-singular since U_1, U_2 are not divisible by $(x - a_i)$ for any singular points $(a_i, 0)$ of X. We use the reduction algorithm to make (U, V) reduced. □

Chapter 3

Computing Square Roots mod p

3.1 Introduction

As we mentioned in the first chapter that there is no known practical deterministic algorithm to find a square root of a number modulo a prime number p. We also mentioned that Shanks' algorithm is the only one that used in practice. In this section we present a new approach for computing square roots and a geometric analogue of Shanks' Algorithm. The efficiency of the new algorithm that we present here depends on the efficiency of performing group operation on the Jacobian group of a singular hyperelliptic curve of genus 1. A hyperelliptic curve with genus one is called an elliptic curve. Since elliptic curves are the main tools for many crypto systems, there have been very efficient methods offered for performing group operation on the group of an elliptic curve. For elliptic curves, the Jacobian group and the group of the points on the curve are isomorphic. The most efficient available algorithm for computing in groups of elliptic curves over finite fields involves only addition and multiplication of numbers in finite fields.[8, Section 13] Since the running time of our algorithm depends on only the running time of an addition algorithm in the group of an elliptic curve, it is expected that our algorithm should be asymptotically faster than Shanks' algorithm.

3.2 A New Algorithm for Computing Square Roots mod p

Since we already have trivial methods for all primes except primes $p \equiv 1 \pmod 8$, we assume all primes $p \equiv 1 (\text{mod } 8)$. Hence $p - 1 = 2^n m$ for some $n \geq 3$ and $(m, 2) = 1$.

In this part we work with some particular singular hyperelliptic curves over a finite field $k = \mathbb{F}_p$. The curves that we use here have only nice singularities, i.e. **nodes**. The general form of a nodal curve in this section is $y^2 = x(x - a_1)^2 \dots (x - a_n)^2$ for distinct a_i's. We show in a moment that the Jacobian group of a nodal curve is isomorphic to the group $\mathbb{G}_{m_1} \oplus \cdots \oplus \mathbb{G}_{m_n}$ where $\mathbb{G}_{m_i}$ is a cyclic subgroup of $\mathbb{F}^*_{p^2}$ of order $p - 1$ or $p + 1$. The main idea of the proof comes from the observation of constructing nodal curves from non-singular curves. For this, consider the curve $\mathbb{P}^1_k$ over a field k. It has $\text{Pic}^o(\mathbb{P}^1_k) = 0$, i.e. every divisor of degree zero is a divisor of a function. Let a_i, b_i for $i = 1, \dots, n$ be n pairs of points on $\mathbb{P}^1_k$. We identify each a_i with the corresponding b_i for $i = 1, \dots, n$. Then we obtain a singular hyperelliptic curve X with only nodes. Note that the normalization X' of X is just $\mathbb{P}^1$. We can perform group operations on $\text{Jac}(X)$ in many different ways[17, section 3.5] or [21, Chapter IV]. However we are going to use the Mumford Representation and Cantor's Algorithm for the group operation as we described in previous chapter.

We begin with the easiest case. Let A and B be two different points on $\mathbb{P}^1$. We denote by X the curve obtained from $\mathbb{P}^1$ by identifying A with B. The canonical map $\pi : \mathbb{P}^1 \to X$ is the normalization morphism of X. By lemma 1.1.4, that induces an exact sequence

$$0 \to \mathcal{O}_X(X) \to \mathcal{O}_{X'}(X') \to \mathcal{S}_A \to 0$$

where the last map $\mathcal{O}_{X'}(X') \to \mathcal{S}_A$ is given by evaluation of a function $f \in \mathcal{O}_{X'}(X')$ at A minus evaluation of it at B. That shows that the regular functions on X are exactly the functions on $\mathbb{P}^1$ that have the same value at A and B. This short exact sequence induces a long exact cohomology sequence

$$0 \to H^o(X, \mathcal{O}_X^*) \to H^o(\mathbb{P}^1, O_{\mathbb{P}^1}^*) \to \mathcal{S}_A^* \to Pic(X) \to Pic(\mathbb{P}^1) \to 0.$$

Since we have $\text{Pic}^o(\mathbb{P}^1) = 0$, $\text{Pic}^o(X) \simeq \mathcal{S}_A^*$. By lemma 1.1.4, we get $\mathcal{S}_A = \left(\dfrac{\mathcal{O}'_{X,A}}{\mathcal{O}_{X,A}}\right)$ where $\mathcal{O}'_{X,A}$ is the integral closure of $\mathcal{O}_{X,A}$ which is equal to $\mathcal{O}_{\mathbb{P}^1,A} \cap \mathcal{O}_{\mathbb{P}^1,B}$. The curve X is analytically isomorphic to the curve $y^2 = x(x-a)^2$, which is a singular hyperelliptic curve of arithmetic genus 1, which is an elliptic curve. The normalization morphism $\pi : \mathbb{P}^1 \to X$ maps two identified points A' and B' in $\mathbb{P}^1$ to the point $(a, 0)$ on X.

We are going to investigate the structure of the Jacobian of a hyperelliptic curve over a finite field. From now on, the ground field k is a finite field with characteristic p different from 2. The $\text{Pic}(X)$ of the singular elliptic curve $X : y^2 = x(x-a)^2$ with $a \neq 0$ is $\mathcal{S}_P^* = \left(\dfrac{\mathcal{O}^{*'}_{X,P}}{\mathcal{O}^*_{X,P}}\right)$ where $P = (a, 0) \in X$ and $\mathcal{O}'_{X,P}$ is the integral closure of $\mathcal{O}_{X,P}$.

Theorem 3.2.1. *Let X be the same as above, i.e. it is a singular hyperelliptic curve defined by an algebraic equation $y^2 = x(x-a)^2$, $a \neq 0$, over a field $k = \mathbb{F}_q$ with $q = p^n$ points. Then Jac(X) is isomporhic to the cyclic multiplicative group F_q^* if a is square in $\mathbb{F}_q$ and isomorphic to a subgroup of $\mathbb{F}_{q^2}^*$ of order $q+1$ if a is not a square in $\mathbb{F}_q$.*

Proof. See [24, Theorem 2.30]. □

Lemma 3.2.2. *Let $E : y^2 = x(x+a)^2$ with $a \neq 0$ be a singular elliptic curve over a field $\mathbb{F}_p$. Then any non-principal divisor class in Jac(E) has a unique representative in $\mathcal{H}_E$ of the form $[(x+a)^2, t(x+a)]$ for some $t \in \mathbb{F}_p$.*

Proof. Let $\mathfrak{A} = \{[(x+a)^2, i(x+a)] \big| \ i \in \mathbb{F}_p$ and $i^2 \neq -a$ if $\sqrt{-a}$ exists in $\mathbb{F}_p$ $\}$ be a subset of $\mathcal{H}_E$. The reduced form of $D_i = [(x+a)^2, i(x+a)]$ is $[(x - i^2), i(i^2+a)]$ so $D_i \neq D_j$ for $i \neq j$ in $\mathbb{F}_p$ which means the inclusion map $g : \mathfrak{A} \to$Jac(E) is injective. Note that D_i is a non-singular element of $\mathcal{H}_E$ unless i is a square root of $-a$ in $\mathbb{F}_p$. Now suppose $-a$ is not a square in $\mathbb{F}_p$. Then by theorem 3.2.1, $\#\text{Jac}(E) = p+1 = \#\Big(\mathfrak{A} \cup \{[1,0]\}\Big)$. Similarly, $\#\text{Jac}(E) = p-1 = \#\Big(\mathfrak{A} \cup \{[1,0]\}\Big)$ if a is a square in $\mathbb{F}_p$. Therefore, the map g is one-to-one and onto. □

Lemma 3.2.3. *Let $E : y^2 = x(x+a)^2$ be the same as above and a be a square $\in \mathbb{F}_p^*$. Then the reduced non-singular pair $[x-a, 2a\sqrt{a}]$ of $\mathcal{H}_E$ corresponds to a divisor class $D \in$ Jac(E) such that D is of order 4. Similarly the divisor class $D_2 \in$ Jac(E) corresponding to $\left[x - \frac{a}{3}, \quad \frac{4a}{3}\sqrt{\frac{a}{3}} \quad \right]$ is of order 3.*

Proof. Let $t \in \mathbb{F}_p$ and $P = [(x+a)^2, t(x+a)]$ be an element of $\mathcal{H}_E$. P is a non-singular element of $\mathcal{H}_E$ unless t is a square root of $-a$. By lemma 3.2.2 any divisor class in Jac(E) has a unique representative in $\mathcal{H}_E$ of the form $[(x+a)^2, t(x+a)]$ for some $t \in \mathbb{F}_p$. Now we use Cantor's Algorithm as described above to compute $2P$.

1. Let say $u_1 = (x+a)^2, \quad u_2 = u_1, \quad v_1 = t(x+a), \quad v_2 = v_1.$

2. gcd$(u_1, u_2, v_1 + v_2)$= gcd$((x+a)^2, (x+a)^2, 2t(x+a)) = (x+a)$

3. $h_1 = 0, h_2 = 0, h_3 = 1/(2t)$

4. $v_0 = \dfrac{(h_1u_1v_2 + h_2u_2v_1 + h_3(v_1v_2 + f))}{(x+a)} = \dfrac{(x(x+a) + t^2(x+a))}{(2t)}$

5. $\widetilde{u} = \dfrac{u_1u_2}{(x+a)^2} = (x+a)^2$

6. $v \equiv v_0(\text{mod } \widetilde{u})$ $v = rem(x(x+a) + t^2(x+a), 2t(x+a)^2) = (rem(x + t^2, 2t(x + a)))(x+a) = \left(\dfrac{t^2 - a}{2t}\right)(x+a)$

7. $u = \widetilde{u}$ and $v = \left(\dfrac{t^2 - a}{2t}\right)(x+a)$ and $2P = [u, v]$

Now P is of order 4 $\Leftrightarrow$ $2P$ is of oder 2 $\Leftrightarrow v = 0 \Leftrightarrow \dfrac{t^2-a}{2t} = 0 \Leftrightarrow t^2 = a$ i.e. P is of order 4 $\Leftrightarrow P = [x - a, 2a\sqrt{a}]$ or $P = [x + a, -2a\sqrt{a}]$ and similarly

1. P is of order 3 $\Leftrightarrow 2P = -P$

2. $2P = \left[(x+a)^2, \left(\dfrac{t^2-a}{2t}\right)(x+a)\right] = [(x+a)^2, -t(x+a)] = -P$

3. $\left(\dfrac{t^2-a}{2t}\right)(x+a) = -t(x+a)$

4. $-t = \dfrac{t^2-a}{2t}$ i.e. $3t^2 = a$.

Hence P is of order 3 if and only if $t = \sqrt{a/3}$ which implies $P = \left[x - \dfrac{a}{3}, \quad \dfrac{4a}{3}\sqrt{\dfrac{a}{3}}\quad\right]$ Since E is an elliptic curve, we can also use standard point addition formula for elliptic curves to prove the lemma. In this case we may represent each divisor class D in Jac(E) by a non-singular point $P = (x, y)$ on E and use point addition method as described in [24, section 2.2]. In a similar way as above one can show that P is of order 4 iff either $P = (a, 2a\sqrt{a})$ or $P = (a, -2a\sqrt{a})$. □

Although it is already known that 3 is not a square for primes $p \equiv 2 \pmod 3$, we show this as a corollary of the previous lemma.

Corollary 3.2.4. 3 *is a quadratic non-residue in* $\mathbb{F}_p$ *where* $p \equiv$ *2 (mod 3) (by default* $p \equiv$ *1 (mod 8)).*

Proof. Assume $p \equiv 2 \pmod 3$ and a is a square mod p which implies $-a$ is also a square. Then the order of singular the elliptic curve $E : y^2 = x(x+a)^2$ is $p-1$. If 3 is a square in $\mathbb{F}_p$, by lemma 3.2.3 we must have a point of order 3. Hence 3 must divide $p-1$ but $p-1 \equiv 1 \pmod 3$. □

We now know 3 is a quadratic non-residue for primes $p \equiv 2 \pmod 3$. Hence Shanks's algorithm finds the square root of a number deterministically for primes $p \equiv 2 \pmod 3$. From now on we may also assume all primes $p \equiv 1 \pmod 3$. Note that we already have $p \equiv 1 \pmod 8$.

Proposition 3.2.5. *Let* $p \equiv 1$ *(mod 8) be a prime number and* $E : y^2 = x(x+a)^2$ *be a singular elliptic curve over a field* $\mathbb{F}_p$ *where* a *is a square mod* p*. Then the probability that a random point* $D \in E$ *is of order divisible by 4 is at least* $3/4$ *and if we also assume* $p \equiv 1$ *(mod 24) then the probability that a random point* $D \in E$ *is of order divisible by 4 or 3 is at least* $11/12$.

Proof. Since a is a square in $\mathbb{F}_p$, the group $\mathrm{Jac}(E)$ is cyclic of order $p-1$ where $p-1 = 2^n 3^m s$ for some non-negative integer $n \geq 3, m \geq 0$ and $(s,6) = 1$. Hence

Jac(E) $\simeq \mathbb{Z}/3^m\mathbb{Z} \oplus \mathbb{Z}/2^n\mathbb{Z} \oplus \mathbb{Z}/t\mathbb{Z}$. Then the probability of the order of D being a multiple of 4 is $(2^n - 2)/2^n$. For the second part i.e. assuming $m \geq 1$, note that Jac(E) has $\mathbb{Z}/24\mathbb{Z}$ as a quotient. So the chance of a random divisor D has order divisible by 4 or 3 is at least 22/24. □

The first algorithm that we see below first searches a point of order divisible by 4. We can say that the chance for a random prime $p \equiv 1 \bmod(8)$ and a random divisor class D in Jac(E) is of order divisible by 4 is 5/6, since

$$\sum_{i=3}^{\infty} \frac{1}{2^{i-2}} \left(\frac{2^i - 2}{2^i}\right) = \sum_{i=3}^{\infty} \frac{1}{2^{i-2}} \left(1 - \frac{2}{2^i}\right)$$

$$= \sum_{i=3}^{\infty} \frac{1}{2^{i-2}} - \frac{8}{4^i} = 1 - 8(1/4^3 + 1/4^4 + 1/4^5 + \dots)$$

$$= 1 - 1/6 = 5/6$$

3.2.1 Algorithm 1

Algorithm 1 for Computing Square Roots:

Input: a number a and an odd prime number p such that a is a quadratic residue mod p and $p - 1 = 4^e m = 2^{e_1} m_1$ and $(m, 4) \neq 4$, $(2, m_1) = 1$ with $e > 1$, $e_1 > 2$.

Output: $\sqrt{a}$ mod p.

Let $E : y^2 = x(x + a)^2$ be a singular elliptic curve over $\mathbb{F}_p$. Note that Jac(E)is cyclic of order $p - 1$. Let P_∞ be the identity element and $P_2 = (0, 0)$ be the point of order 2 in Jac(E). We do

repeat :

1. pick a random point $P = (x, y)$ on E.

```
2. Compute $Q = mP$.
   **until** $Q$ is not $P_\infty$ or $P_2$ in $E$.
   **repeat :**
3. $Q_1 = 2^i Q$ for $i = 0, \ldots, e_1 - 1$.
4. Compute $Q = Q_1 = (z, w)$
   **until** $z = a$
5. compute $w/2a$ which gives $\sqrt{a}$ mod $p$
```

It is easy to show that, steps 2 and 3 of the algorithm 1 requires at most $2lgp$ steps consisting of doubling and addition in the group Jac(E). If we use projective coordinates each doubling or addition costs approximately $12lgp$ steps [8, Chapter 13]. Overall the expected running time for computing $\sqrt{a}$ mod p is $O(lg^2 p)$.

If the point Q in the step 2 is not P_∞ or P_2, then its order must be a multiple of 4. That means a power of Q of the form 2^i must be a point of order 4. Hence, if we must get $z = a$ in the second part of the algorithm if the order of Q is a multiple of 4 by lemma 3.2.3. Therefore we reach the point $Q = (a, \mp 2a\sqrt{a})$. The first part of algorithm 1 searches a point of order multiple of 4 on E. The probability that a random point P on E is of order divisible by 4 for a random prime p is 5/6 by proposition 3.2.5. For a fixed prime p such that $p - 1 = 2^n m_1$, $(m_1, 2) = 1$, the chance is $(2^n - 2)/2^n$. Therefore, in practice, we will find such a point P very quickly. For example the probability that one can find such a P after 4 steps is more than 99/100. We may modify the algorithm by searching for a point P of order a multiple of not just 4 but also 3. This increases the chance of success at the first part to at least 11/12 for a point P. However in this case we also need to know $\sqrt{3}$ which can be computed by Schoof's algorithm. Although we will have the same the asymptotic running time, Schoof's algorithm increases the heuristic running time.

3.3 A Geometric Analogue of Shanks' Algorithm

Now we are going to use nodal curves with larger arithmetic genus to describe a method to compute square roots. This method will be a geometric interpretation of Shanks' method. Now pick 2 pairs of points $a_i, b_i \in \mathbb{P}^1$ and identify each a_i with b_i. We get a nodal curve X with only two ordinary singularities such that the canonical map $\pi : X' = \mathbb{P}^1 \to X$ is the normalization morphism of X. Using similar notations as above, we have an exact sequence

$$0 \to \mathcal{O}_X(X) \to \mathcal{O}_{X'}(X') \to \bigoplus_{a_i} \mathcal{S}_{a_i} \to 0$$

This short exact sequence induces a long cohomology sequence so that we have $\mathrm{Pic}(X) \simeq \mathcal{S}^*_{a_1} \oplus \mathcal{S}^*_{a_2}$. Now consider the nodal curve $H : y^2 = x(x-a)^2(x-b)^2$ over a field $\mathbb{F}_p$ for non-zero distinct a and b. The curve H is analytically isomorphic to the curve X. Let $\pi : \mathbb{P}^1 \to H$ be the normalization morphism such that $\pi^{-1}(p_1) = \{a_1, b_1\}$ and $\pi^{-1}(p_2) = \{a_2, b_2\}$, where $p_1 = (a, 0)$, $p_2 = (b, 0)$. Now as in the case of singular elliptic curve we have

Theorem 3.3.1. *Let H be the singular hyperelliptic curve defined by $y^2 = x(x-a)^2(x-b)^2$ with $a, b \neq 0$ over a field $k = \mathbb{F}_q$ with $q = p^n$ points. Then Jac(H) is isomorphic to the group $\mathbb{G}_a \oplus \mathbb{G}_b$ such that $\mathbb{G}_a$ (or $\mathbb{G}_b$) $\simeq \mathbb{F}^*_q$ if a (or b) is a square* *and $\mathbb{G}_a$ (or G_b) isomorphic to a subgroup of $\mathbb{F}_{q^2}^*$ of order $q+1$ if a (or b) is not a square in $\mathbb{F}_q$.*

Proof. Let $p_1 = (a, 0), p_2 = (b, 0)$ be the singular points on H. The discussion above shows that $\text{Jac}(H) \simeq \mathcal{S}_{p_1}^* \oplus \mathcal{S}_{p_2}^*$ where $\mathcal{S}_{p_i} = \left(\dfrac{\mathcal{O}'_{H,p_i}}{\mathcal{O}_{H,p_i}}\right)$ and $\mathcal{O}'_{H,p_i}$ is the integral closure of $\mathcal{O}_{H,p_i}$. From theorem 3.2.1, $\mathcal{S}_{p_1}^*$(or $\mathcal{S}_{p_2}^*$) is isomorphic to $\mathbb{F}_q^*$ if a (or b) is a square and a subgroup of $\mathbb{F}_{q^2}^*$ of order $q+1$ if a (or b) is not a square in $\mathbb{F}_q$. □

Proposition 3.3.2. *Let H be a hyperelliptic curve defined by $y^2 = (x)(x^2 - a)^2$ over a field $\mathbb{F}_q$ and a be a square in $\mathbb{F}_q$. Then for any $t \neq \sqrt{a} \in \mathbb{F}_q^*$, $D = [(x^2 - a)^2, t(x^2 - a)] \in Jac(H)$ is the sum of $D_1 = [(x - \sqrt{a})^2, m(x - \sqrt{a})]$ and $D_2 = [(x + \sqrt{a})^2, -m(x + \sqrt{a})]$ for some $m \in \mathbb{F}_q^*$.*

Proof. We need to check first if $[(x^2 - a)^2, t(x^2 - a)]$ is non-singular in $\mathcal{H}_H$, i.e. whether $t^2 = \sqrt{a}$. Now assume it is non-singular in $\mathcal{H}_H$ and let $D_1 = (U_1, V_1) = [(x - \sqrt{a})^2, m(x - \sqrt{a})]$, and $D_2 = (U_2, V_2) = [(x + \sqrt{a})^2, -m(x + \sqrt{a})]$. We use Cantor's Algorithm to find $D_1 + D_2$.

1. $\gcd(U_1, U_2, V_1 + V_2(= -2m\sqrt{a})) = 1$ and $h_1 = 0, h_2 = 0$ and $h_3 = \dfrac{1}{-2m\sqrt{a}}$.
2. $V_0 = (U_1V_2h_1 + U_2V_1h_2 + (V_1V_2 + f)h_3)/d = (-m^2(x^2-a) + x(x^2-a)^2)/(-2m\sqrt{a})$.
3. $U = (x^2 - a)^2$ and $V \equiv V_0 \equiv \dfrac{m}{2\sqrt{a}}(x^2 - a) \bmod ((x^2 - a)^2)$.

Hence for $m = (2\sqrt{a})t$ we have $D_1 + D_2 \simeq [(x^2 - a)^2, t(x^2 - a)] \simeq D$. □

Corollary 3.3.3. *Let H be the same as above and $p \equiv 3(mod\ 4)$. Suppose a is a square in $\mathbb{F}_q$. Let D be a divisor in $Jac(H)$ corresponding to a non-singular pair*

$[(x^2-a)^2,(x^2-a)]$. *Then* $(q-1)D$ *corresponds to* $[(x+\sqrt{a})^2, r(x+\sqrt{a})]$ *for some* $r \in \mathbb{F}_q$ *if* $\sqrt{a}$ *is a square in* $\mathbb{F}_q$ *otherwise* $[(x-\sqrt{a})^2, m(x-\sqrt{a})]$ *for some* $m \in \mathbb{F}_q$ *.*

Proof. Since $p \equiv 3 \pmod 4$, $\left(\frac{\sqrt{a}}{p}\right) = -\left(\frac{-\sqrt{a}}{p}\right)$. Hence the order of $\mathrm{Jac}(H)$ is $(p-1)(p+1)$. Suppose $\sqrt{a}$ is a square mod p. Let $p_1 = (\sqrt{a},0)$, $p_2 = (-\sqrt{a},0)$ be the singular points of H. We know $\mathrm{Pic}^o(H) \cong \frac{\mathcal{O}'_{H,p_1}}{\mathcal{O}_{H,p_1}} \oplus \frac{\mathcal{O}'_{H,p_2}}{\mathcal{O}_{H,p_2}}$. Let $D_1 = ((x-\sqrt{a})^2, 2\sqrt{a}(x-\sqrt{a})$ and $D_2 = ((x+\sqrt{a})^2, 2\sqrt{a}(x+\sqrt{a})$ be divisors in $\mathrm{Jac}(H)$. We have $D_1 \in \frac{\mathcal{O}'_{H,p_1}}{\mathcal{O}_{H,p_1}}$ and $D_2 \in \frac{\mathcal{O}'_{H,p_2}}{\mathcal{O}_{H,p_2}}$, so the order of D_1 divides $p-1$ and the order of D_2 divides $p+1$. Then by Proposition 3.3.2, $D = D_1 + D_2$. Therefore $(p-1)D = (p-1)D_1 + (p-1)D_2 = (p-1)D_2 = ((x+\sqrt{a})^2, r(x+\sqrt{a}))$ for some $r \in \mathbb{F}_q$. Note that any multiple of D_2 is of the form $((x+\sqrt{a})^2, l(x+\sqrt{a}))$ for some $l \in \mathbb{F}_p$. □

3.3.1 Algorithm 2

input: A square number a in $\mathbb{F}_p$.

output: $\sqrt{a} \in \mathbb{F}_p$.

Let $H : y^2 = x(x^2-a)^2$ be a singular hyperelliptic curve over $\mathbb{F}_p$ and $p \equiv 3 \pmod 4$.

1. pick a random $D = [(x^2-a)^2, t(x^2-a)] \in \mathrm{Jac}(H)$ for some $t \in F_p^*$.

2. compute $(p-1)D$.

By previous corollary, the output at step 2 is $[(x-\sqrt{a})^2, n(x-\sqrt{a})]$ for some $n \in \mathbb{F}_p^*$ hence the algorithm returns a $\sqrt{a} \in \mathbb{F}_p$. This method can be considered a geometric analogue of the well-known trivial method that we described in the first chapter.

3.3.2 Algorithm 3

Proposition 3.3.4. *Let $p \equiv 1$ $(mod\ 8)$ be a prime number and $H: \quad y^2 = x(x-(b+\sqrt{a})^2)(x-(b-\sqrt{a}))^2 = x(x^2-2bx+b^2-a)^2$ be a singular hyperelliptic curve over a field $\mathbb{F}_p$. Assume a is a square. If b^2-a is not a square in $\mathbb{F}_p$ we can find a square root of a mod p by computing $(p-1)D = (p-1)[(x^2-2bx+b^2-a)^2, (x^2-2bx+b^2-a)]$ in Jac(H).*

Proof. Since b^2-a is quadratic non-residue mod p and a is a square, $\left(\dfrac{b-\sqrt{a}}{p}\right) = -\left(\dfrac{b+\sqrt{a}}{p}\right)$. WLOG we may assume $b-\sqrt{a}$ is a square mod p. Then $\#\text{Jac}(H) = (p-1)(p+1)$. We know $D = [(x^2-2bx+b^2-a)^2, (x^2-2bx+b^2-a)] = D_1 + D_2$ for some $D_1 = [(x-(b-\sqrt{a}))^2, t_1(x-(b-\sqrt{a}))]$ and $D_2 = [(x-(b+\sqrt{a}))^2, t_2(x-(b+\sqrt{a}))]$. D_1 has order dividing $p-1$ and D_2 has order dividing $p+1$. Hence $(p-1)D = (p-1)D_1 + (p-1)D_2 = (p-1)D_2 = [(x-(b+\sqrt{a}))^2, r(x-(b+\sqrt{a}))]$ for some $r \in \mathbb{F}_p$. □

Let us assume a is a square mod p. Now pick a random number $b \in \mathbb{F}_p$ and consider the hyperelliptic curve $H: y^2 = x(x-(b+\sqrt{a}))^2(x-(b-\sqrt{a}))^2 = x(x^2-2bx+b^2-a)^2$ over a field $\mathbb{F}_p$ and $p \equiv 1$ (mod 8). The probability that b^2-a is a quadratic non-residue mod p is 1/2. In this case for a random point

$D = [(x^2 - 2bx + b^2 - a)^2,\ t(x^2 - 2bx + b^2 - a)]$ in $\mathrm{Jac}(H)$, $(p-1)D$ is equal to $[(x-(b \mp \sqrt{a}))^2, r(x-(b \mp \sqrt{a}))]$ for some $r \in \mathbb{F}_p$. Suppose $b^2 - a$ is a square mod p. We now explain in this case we may still find $\sqrt{a}$. We can easily determine the order of $\mathrm{Jac}(H)$ which is either $(p-1)^2$ or $(p+1)^2$. We have three points in $\mathrm{Jac}(H)$ of order 2. They are $P_0 = [x, 0]$, $P_1 = [(x-(b-\sqrt{a}))^2, 0]$ and $P_2 = [(x-(b+\sqrt{a}))^2, 0]$. The probability that D is of even order is at least $3/4$. Now assume D is of even order and $\#\mathrm{Jac}(H) = 2^e m$ such that $e > 0$ and $(m, 2) = 1$. Then $(2^i m D)$ is of order 2 for some $i = 0, \dots, e-1$ and the chance that it is either P_1 or P_2 is $2/3$. Since, as explained above, we already have $1/2$ chance to reach $\sqrt{a}$ by computing the $(p-1)$th power of D, overall we have at least $\frac{1}{2} + \frac{1}{2} \cdot \frac{3}{4} \cdot \frac{2}{3} = \frac{3}{4}$ chance to find a square root of a mod p by using a single divisor class $D \in \mathrm{Jac}(H)$. We have just justified the following algorithm.

Algorithm 3:

Input: A square number a mod p where $p \equiv 1 \pmod 8$

Output: A square root of a mod p

1. Pick a random number b mod p and let $H: \quad y^2 = x(x^2 - 2bx + b^2 - a)^2$
2. Compute $(p-1)D$ in $\mathrm{Jac}(H)$ where $D = [(x^2 - 2bx + b^2 - a)^2, (x^2 - 2bx + b^2 - a)]$
3. if $(p-1)D$ is of the form $[(x-(b \pm \sqrt{a}))^2, t_1(x-(b \pm \sqrt{a}))]$ return $\sqrt{a}$ otherwise i.e. $[1, 0]$ go to step 4
4. Determine e such that $p - 1 = 2^e v$ with $(v, 2) = 1$
5. Compute $\widetilde{D} = vD$

```
6. if $\widetilde{D} = [1, 0]$ go to step 1 otherwise do
   **repeat:**
7. $\widetilde{\widetilde{D}} = 2^s\widetilde{D}$ for $e = 0, \ldots, e - 1$
8. Compute $\widetilde{D} = \widetilde{\widetilde{D}}$
   **until:** $\widetilde{D} = [h(x), 0]$
9. if $h(x) = x$ go to step (1) otherwise return $\sqrt{a}$
```

Since addition of the points in $\mathrm{Jac}(H)$ requires polynomial gcd, the expected running time of algorithm 3 is not better than algorithm 1. Although this can be considered a geometric analogue of Shanks-Tonelli's algorithm, we have greater chance to find $\sqrt{a}$ mod p in a single trial. Actually, as we showed, the probability that the algorithm returns a square root in a single trial is 3/4. The asymptotic running time of this algorithm is the same as Shanks-Tonelli's algorithm which is $O(lg^3p)$.

Chapter 4

Factorization of Polynomials mod p

4.1 Introduction

We described the Mumford Representation and Cantor's Algorithm for singular hyperelliptic curves in the second chapter. The efficiency of Cantor's Algorithm makes hyperelliptic curves suitable for many applications. There are many crypto systems whose key ingredient is a hyperelliptic curve. In this chapter we investigate another application of hyperelliptic curves especially singular hyperelliptic curves. This application is factorization of polynomials over finite fields. The methods, which we describe in this chapter, for polynomial factorization can also be used with non-singular hyperelliptic curves. However we will see that in this case one needs to know the order of their Jacobians and at least one non-trivial element of them, which requires tedious work for especially higher genus. On the contrary we have many trivial tools to determine the order of a singular hyperelliptic curve's Jacobian and a point in it. We describe them later in this chapter.

One of the main ideas that we use for factorization of polynomials can be illustrated as follows. Let $f(x)$ be a monic square-free polynomial of degree 3 with coefficients in a field k. Consider the non-singular hyperelliptic(elliptic) curve X of arithmetic genus 1 defined by the algebraic equation $y^2 = f(x)$. The Jacobian, $\mathrm{Jac}(X)$, of the elliptic curve X is the same group as the group of the points on the elliptic curve. Namely each point (x, y) on the curve represents a unique divisor class in $\mathrm{Jac}(X)$. Suppose the group $\mathrm{Jac}(X)$ has even order. This means there exists a point (a, b) of order 2, i.e. $b = 0$ and $(x - a)$ is a factor of $f(x)$. Hence the problem of factoring $f(x)$ is reduced to determining an element of order 2 in $\mathrm{Jac}(X)$. Assume a point $P = (x, y)$ is of even order on the curve and $\#\mathrm{Jac}(X) = 2^m(2n + 1)$ for $m \neq 0, n \in \mathbb{Z}$. Then the $2^i(2n + 1)^{th}$ power of P gives us a point of order 2 for some $0 \leq i < m$. If $\mathrm{Jac}(X)$ is of odd order, we conclude that $f(x)$ is an irreducible polynomial in $k[x]$. We extend this idea to use for factorization of polynomials of any degree.

We described the well-known algorithms for polynomial factorization mod p in chapter 1. We now generalize the ideas of the previous section to find a factor of a polynomial over a finite field $k = \mathbb{F}_p$ with p elements where p is a prime number. Consider a square-free polynomial $f(x)$ of degree n in $k[x]$. We know that the polynomial $f(x)$ is irreducible if and only if $f(x)$ divides $x^{p^n} - x$ and is coprime to the polynomial $x^{p^{n/q}} - x$ for each prime q divides n [9, Proposition 3.4.4]. Hence, the primality test for $f(x)$ can be done in a reasonable amount of time, so we assume $f(x)$ is not an irreducible polynomial throughout this chapter.

In the previous chapter we used nodal curves with at most two singular points for computing square roots. Now we are going to use nodal curves with any number of singularities for factorization of polynomials. The Jacobian of such a curve is always a direct sum of cyclic subgroups of the multiplicative group of a field which is a finite extension of k.

4.2 Computing in Jacobians of Nodal Curves

Let $k = \mathbb{F}_p$ be a finite field with p elements. In this section we investigate group operation and the representation of group elements of the Jacobian of a nodal curve of the form $y^2 = xf(x)^2$ for a square-free polynomial $f(x) \in k[x]$.

Proposition 4.2.1. *Let $f(x)$ be a reducible square-free polynomial of degree n in $k[x]$ and $q_1(x)$, $q_2(x)$ be non-constant polynomials in $k[x]$ such that $q_1(x)q_2(x)$ divides $f(x)$ and $(q_1(x)$,$q_2(x))$=1. Let $H : y^2 = xf^2(x)$ be a singular hyperelliptic curve over k. Suppose the divisor class $D_q \in$Jac(H) corresponds to the non-singular pair $[q_1^2(x)q_2^2(x),\ t(x)q_1(x)q_2(x)]$ in $\mathcal{H}_H$ for some polynomial $t(x) \in k[x]$. Then*

$$D_q = [q_1^2(x)q_2^2(x), t(x)q_1(x)q_2(x)] = [q_1^2(x), \widetilde{h}_1(x)q_1(x)]+[q_2^2(x), \widetilde{h}_2(x)q_2(x)] = D_{i_1}+D_{i_2}$$

for some polynomials $\widetilde{h}_1(x)$, $\widetilde{h}_2(x)$ in $k[x]$.

Proof. Consider divisor classes $D_{i_1} = [u_1, v_1] = [q_1^2, t(x)q_1(x)q_2(x)]$ and $D_{i_2} = [u_2, v_2] = [q_2^2, t(x)q_1(x)q_2(x)]$. Although we may have $\deg(u_i) \leq \deg(v_i)$, we can see that the invertible $\mathcal{O}_H$-modules $I = (u_i, y - v_i)$ and $J = (u_i, y - (v_i(\text{mod } u_i)))$ are the same for $i = 0, 1$. Hence $[u_i, v_i(\text{mod } u_i)]$ represents the same divisor class as $[u_i, v_i]$ for $i = 0, 1$. Since the pair $[q_1^2(x)q_2^2(x), t(x)q_1(x)q_2(x)]$ is non-singular in $\mathcal{H}_H$, $[q_1(x)^2, t(x)q_1(x)q_2(x)]$ and $[q_2(x)^2, t(x)q_1(x)q_2(x)]$ are also non-singular. Now we compute $D_{i_1} + D_{i_2}$ by using Cantor's Algorithm

1. $\gcd(q_1^2, q_2^2, t(x)q_1(x)q_2(x) + t(x)q_1(x)q_2(x)) = \gcd(q_1^2, q_2^2) = 1$

2. $(r_1(x)q_1(x)^2 + r_2(x)q_2(x)^2 + r_3(x)(2t(x)q_1(x)q_2(x)) = 1$ for some polynomials

$r_1(x)$, $r_2(x)$ and $r_3(x) = 0$

3. Then $u = q_1^2 q_2^2$ and

4. $v = r_1 q_1^2 t q_1 q_2 + r_2 q_2^2 t q_1 q_2 + (r_3)(t^2 q_1^2 q_2^2 + x f^2) = t q_1 q_2 (r_1 q_1^2 + r_2 q_2^2) = t(x) q_1(x) q_2(x)$

5. hence $D_{i_1} + D_{i_2} = [u, v] = D_q$

□

Corollary 4.2.2. *Let $f(x) = f_1(x) \dots f_m(x)$ and $f_1(x), \dots, f_m(x)$ be the irreducible factors of $f(x)$. Let $H : y^2 = xf(x)^2$ be a singular hyperelliptic curve and D be the divisor class in Jac(H) corresponding to the non-singular pair $[f(x)^2, h(x)f(x)]$ for some polynomial $h(x) \in k[x]$ such that deg($h(x)$) <deg($f(x)$). Then,*

$$D = [f(x)^2, g(x)f(x)] = [f_1^2(x), h_1(x)f_1(x)] + \cdots + [f_m^2(x), h_m(x)f_m(x)] = D_1 + \cdots + D_m$$

for some polynomials $h_j(x)$ with deg($h_j(x)$) <deg($f_j(x)$) for $j = 0, \dots, m$.

Proof. By using the above proposition with induction on m, we get the result. □

Lemma 4.2.3. *Let $f(x) \in k[x]$ be a reducible square-free polynomial such that $f_1(x) \in k[x]$ is an irreducible factor of it. Let $H : y^2 = xf^2(x)$ be a hyperelliptic curve over k. Assume deg($f_1(x)$) $= d_1$ and $\mathcal{F}$ is the set of the divisor classes in Jac(H) corresponding to the non-singular pairs of the form $[f_1(x)^2, t(x)f_1(x)]$ such that $t(x) \in k[x]$ with deg($t(x)$) $< d_1$. Then $\mathcal{F}$ is a subgroup of Jac(H) and the order of any element of it divides either $p^{d_1} - 1$ or $p^{d_1} + 1$.*

Proof. Any pair of the form $[f_1(x)^2, t(x)f_1(x)]$ is non-singular iff $\gcd(x-t(x)^2, f_1(x)) = 1$. We claim that for any two divisor classes $D_1 = [f_1(x)^2, t_1(x)f_1(x)]$ and $D_2 = [f_1(x)^2, t_2(x)f_1(x)]$ where $f_1(x)$ is an irreducible factor of $f(x)$, $D_1 + D_2$ is either $[f_1(x)^2, t_3(x)f_1(x)]$ in $\mathcal{F}$ or [1,0]. By applying Cantor's Algorithm on D_1 and D_2, we have

1. $g(x)$=gcd$(f(x)_1^2, f(x)_1^2, (t_1(x)+t_2(x))f_1(x)) = f_1(x) \cdot gcd(f_1(x), t_1(x)+t_2(x))$= $f_1(x)$ or $f_1(x)^2$, since $f_1(x)$ is an irreducible polynomial.

2. if $g(x) = f_1(x)^2$ then $u(x) = 1$ otherwise $u(x) = f_1(x)^2$.

3. if $u(x) = f_1(x)^2$, then $v(x) = t_3(x)f_1(x)$ for some polynomial $t_3(x) \in k[x]$ of degree less than d_1, since $(v(x)^2 - xf(x))$ is a multiple of $u(x)$.

4. Therefore $D_1 + D_2 = [f_1(x)^2, t_3(x)f_1(x)] \in \mathcal{F}$ or $[1, 0]$

This shows that $\mathcal{F}$ is a subgroup of Jac(H).

Now we show that any element of $\mathcal{F}$ divides either $p^{d_1} - 1$ or $p^{d_1} + 1$. The roots of $f_1(x)$ are in $\mathbb{F}_{p^{d_1}}$. Suppose $f_1(x) = (x - \alpha_1) \ldots (x - \alpha_{d_1})$ in $\mathbb{F}_{p^{d_1}}[x]$ and consider the curve H over $\mathbb{F}_{p^{d_1}}$. Then the set $\mathcal{F}_{\alpha_i}$ consisting of reduced divisors of the form $[(x - \alpha_i)^2, \beta(x - \alpha_i)]$ for $\beta \in \mathbb{F}_{p^{d_1}}$ is a subgroup of Jac$(H_{\mathbb{F}_{p^{d_1}}})$, since $f_{\alpha_i} = x - \alpha_i$ is an irreducible factor of $f(x)$ in $\mathbb{F}_{p^{d_1}}[x]$. The reduced pair $[(x - \alpha_i)^2, \beta(x - \alpha_r)]$ is non-singular iff $\beta \neq \sqrt{\alpha_i}$. Hence the order of subgroup $\mathcal{F}_{\alpha_i}$ of Jac$(H_{\mathbb{F}_{p^{d_1}}})$ is $p^{d_1} - 1$ if $\sqrt{\alpha_i}$ exists in $\mathbb{F}_{p^{d_1}}$ and $p^{d_1} + 1$ otherwise. By proposition 4.2.1, $[f_1(x)^2, t(x)f_1(x)]$ in Jac$(H_{\mathbb{F}_{p^{d_1}}})$ is a sum of divisors of the form $D_i = [(x - \alpha_i)^2, \beta_i(x - \alpha_i)]$ for some $\beta_i \in \mathbb{F}_{p^{d_1}}$. Then consider

$$(p^{d_1}-1)[f_1(x)^2, t(x)f_1(x)] = (p^{d_1}-1)\Big(D_i + \cdots + D_{d_1}\Big) =$$
$$(p^{d_1}-1)\Big([(x-\alpha_i)^2, \beta_i(x-\alpha_i)] + \cdots + [(x-\alpha_{d_1})^2, \beta_{d_1}(x-\alpha_{d_1})]\Big).$$

Since the the pair $[f_1(x)^2, t(x)f_1(x)]$ corresponds to a divisor class in Jac($H_{\mathbb{F}_p}$), then $(p^{d_1}-1)[f(x)^2, g(x)f(x)]$ is in Jac($H_{\mathbb{F}_p}$). Suppose only some D_i disappear in this operation. Then by Cantor's Algorithm, the result must be of the form $[\widetilde{f_1}(x)^2, h_1(x)]$ where $\widetilde{f_1}(x)$ is a non-trivial factor of $f_1(x)$ and $h(x)$ is a polynomial. Since $f_1(x)$ is an irreducible polynomial in $\mathbb{F}_p[x]$, it is impossible that only some D_i disappear. That means either all D_i's disappear or none. Hence the order of each D_j divides either $p^{d_1}+1$ or $p^{d_1}-1$. Therefore the order of $[f_1(x)^2, t(x)f_1(x)]$ divides either $p^{d_1}+1$ or $p^{d_1}-1$. □

Proposition 4.2.4. *Let $f(x)$ be a square-free reducible polynomial of degree n in $k[x]$ and $H : y^2 = xf(x)^2$ be a singular hyperelliptic curve over k. Suppose the divisor class D in Jac(H) corresponds to the non-singular pair $[f(x)^2, h(x)f(x)]$ for some polynomial $h(x) \in k[x]$ such that deg($h(x)$) <deg($f(x)$). Then the ($p^i \pm 1$) power of D for $i = 1, \ldots, \widetilde{d}$ gives either a factor of $f(x)$ or the pair $[1, 0]$ where $\widetilde{d}$ =max$\{d_j$ =degree of a irreducible factor of $f(x)\}$.*

Proof. Suppose the irreducible factors of $f(x)$ are $f_1(x), \ldots, f_m(x)$ with deg$(f_j(x)) = d_j$ for $j = 0, \ldots, m$. By proposition 4.2.1, we have

$$D = [f(x)^2, h(x)f(x)] = [f_1^2(x), h_1 f_1(x)] + \cdots + [f_m^2(x), h_m(x)f_m(x)] = D_1 + \cdots + D_m$$

for some polynomials $h_j(x)$ with deg($h_j(x)$) <deg($f_j(x)$) for $j = 0, \ldots, m$. From

lemma 4.2.3, the order of D_j divides either $p^{d_j}+1$ or $p^{d_j}-1$. Hence the $(p^i \pm 1)$th power of D for some $i = 1, \ldots, \widetilde{d}$ =max$\{d_j\}$ annihilates either some of D_j's or or all of D_j's .

We now show that if the rth of power of D for some $r \in \mathbb{Z}$ annihilates only some of D_j's, we get a non-trivial factor of $f(x)$. Now

$$rD_j = r \cdot [f_j^2(x), h_j(x)f_j(x)] = [f_j^2(x), \widetilde{h}_j(x)f_j(x)]$$

for some polynomial $\widetilde{h}_j(x)$ with deg($\widetilde{h}_j(x)$) <deg($f_j(x)$), since by lemma 4.2.3, the set of the divisor classes in Jac(H) corresponding to the non-singular pairs of the form $[f_j(x)^2, t(x)f_j(x)]$ is a subgroup of Jac(H). We also see that if $f_l(x)$ and $f_s(x)$ are relatively prime factors of $f(x)$,

$$[f_l^2(x), h_l(x)f_l(x)] + [f_s^2(x), h_s(x)f_s(x)] = [f_l^2(x)f_s^2(x), h_{ls}(x)f_l(x)f_s(x)]$$

for some polynomial $h_{ls}(x)$ with degh_{ls} <deg$f_l(x)$+deg$f_s(x)$.

Now suppose $rD = r \cdot (D_1 + \cdots + D_j \cdots + D_m)$ annihilates $\{D_{j_{s_1}}, \ldots, D_{j_{s_r}}\}$ but do not annihilate $\{D_{j_1}, \ldots, D_{j_b}\}$. Then

$r \cdot D = r \cdot \Big([f_1^2(x), h_1 f_1(x)] + \cdots + [f_m^2(x), h_m(x)f_m(x)]\Big)$

$= \Big([f_{j_1}^2(x), \widetilde{h}_{j_1} f_{j_1}(x)] + \cdots + \cdots + [f_{j_b}^2(x), \widetilde{h}_{j_b}(x)f_{j_b}(x)]\Big)$

$= [f_{j_1}^2(x) \ldots f_{j_b}^2(x), h(x)] = \widetilde{D}$ for some polynomial $h(x)$. The first component of $\widetilde{D}$ is a square of non-trivial factor of $f(x)$. □

4.2.1 An Addition Algorithm

Let $f(x)$ be a square-free reducible polynomial with degree n in $k[x]$, H and D be the same as above. In order to find a factor of $f(x)$, proposition 4.2.4 suggests that we need to compute some certain powers of D. To compute a power of D, we perform the addition operation for divisors of the form $[f^2, gf]$. We can do this by using the standard methods, i.e. Cantor's Algorithm with the Reduction algorithm. However the following addition algorithm shows that we do need to use the reduction algorithm for addition operation of these kinds of divisors.

Addition Algorithm:

input: Divisor classes $D_i = [f^2, g_i f]$ in $\mathrm{Jac}(H)$ for some polynomials g_i of degree less than n for $i = 1, 2$ where H is the curve defined by $y^2 = xf(x)^2$.

output: The algorithm returns $D_s = D_i + D_j = [f^2, g_i f] + [f^2, g_j f]$ or a non-trivial factor of $f(x)$.

We do

1. $\gcd(f, g_i + g_j) = r(x)$. If $r(x) \neq 1$, $r(x)$ is a factor of $f(x)$ otherwise do
2. find h_i, h_j such that $h_i f + h_j(g_i + g_j) = 1$
3. $u = f^2$ and $g_s \equiv (fh_i g_i + h_j(g_i g_j + x))(\bmod f)$ with $\deg(g_s) < n$
4. $v = g_s f$ and $D_s = [u, v]$.

Proof. Now we justify the addition algorithm by applying Cantor's Algorithm to $D_i + D_j$

1. $\gcd(f^2, f^2, g_i f + g_j f) = f(x) \cdot \gcd(f, f, g_i + g_j)$

   $= f(x) \cdot \gcd(f, g_i + g_j) = f(x) \cdot r(x)$ with h_i, h_j such that $h_i f + h_j(g_i + g_j) = r(x)$

2. If $r(x) \neq 1$, $r(x)$ is a factor of $f(x)$ otherwise do

3. $u_1 = \dfrac{f^2 f^2}{f^2} = f^2$ and

   $$v_0 = \frac{(h_i f^3 g_i + h_j(g_i g_j f^2 + x f^2))}{f} = h_i f^2 g_i + (h_j)(g_i g_j f + x f)$$

4. $v_1 \equiv v_0 \bmod(f^2) \equiv h_i f^2 g_i + h_j(g_i g_j f + x f)(\text{mod} u_1 = f^2)$

   $= \Big((f h_i g_i + h_j(g_i g_j + x))(\text{mod} f)\Big) f = \Big(g_s\Big) f$ with $\deg(g_s) < n$

5. $D_i + D_j = [u_1, v_1] = [u, v] = D_s$

□

The above algorithm is basically an application of the fact that any a power of $D = [f^2, gf]$ is again of the form $[f^2, g_i f]$ for some polynomial $g_i(x)$. This is the reason that we do need to the use reduction algorithm while computing a power of D.

In the real time implementation performing addition for the divisors of the form $[f^2(x), g(x)f(x)]$ by using the addition algorithm is at least twice as fast in some situations because we are not using the reduction algorithm. However, if $\deg(f(x))$ is large, using the reduction algorithm makes Cantor's Algorithm more efficient.

4.3 An Algorithm for Polynomial Factorization in Finite Fields

Now we are going to use above ingredients to construct a method for polynomial factorization over finite fields. Note that we have

$$D = [f(x)^2, h(x)f(x)] = [f_1^2(x), h_1 f_1(x)] + \cdots + [f_m^2(x), h_m(x)f_m(x)] = D_1 + \cdots + D_m$$

where $f_j(x)$ are irreducible factors of $f(x)$ and $h_j(x)$ are polynomials in $k[x]$ with deg($h_j(x)$) <deg($f_j(x)$) for $j = 1, \ldots, m$. In order to find a factor of $f(x)$, we need to compute $(p^i \pm 1)D$ for $i = 0, \ldots, \widetilde{d}$ where $\widetilde{d}$ =max{deg$f_j(x)$}. Then by proposition 4.2.4, we get either a factor of $f(x)$ or $[1, 0]$. Suppose $(p^s \pm 1)D = [1, 0]$ for some $s = 0, \ldots, \widetilde{d}$. This occurs with the probability $1/2^m$ where m is the number of factors of $f(x)$ since it means that orders of all D_j's divide $(p^s \pm 1)$. Hence this is very unlikely if $f(x)$ has more than 3 factors. However, even in this case we have still a big chance to find a factor of $f(x)$.

Now let us assume that $(p^{d_1} - 1)D = [1, 0]$ and $p^{d_1} - 1 = 2^{e_1}k_1$ for some integer $e_1 > 0$ and odd integer k_1. If the order of D is even, $(2^j k_1)D$ gives us an element of order 2 in Jac(H) for some $j = 0, \ldots, e_1 - 1$. The detailed analysis of the representation of 2-torsion points shows that the elements of order 2 in Jac(H) must be of the form $[x, 0]$, $[f_j^2(x), 0]$, $[xf_j^2(x), 0]$ for some non-trivial factors $f_j(x)$ of $f(x)$. Hence $(2^j k_1)D$ gives either a factor of $f(x)$ or $[x, 0]$. The probability that order of D is even and its $(2^j k_1)^{th}$ power gives a factor of $f(x)$ is $(2^m - 2)/2^m$ where m is the number of factors of $f(x)$. We have just verified the following algorithm :

Factorization Algorithm :

input: A square free polynomial $f(x)$ in $k[x]$ with $\deg(f(x)) = n$, where $k = \mathbb{F}_p$.

output: A non-trivial factor of $f(x)$ in $k[x]$.

Let H be the hyperelliptic curve defined by $y^2 = xf^2(x)$ over $k[x]$. The algorithm stops once it finds a non-trivial factor of $f(x)$.

1. pick a random polynomial $g(x)$ in $k[x]$ such that $\deg(g(x)) < n$.

2. compute $r(x)$=gcd$(g(x), f(x))$, if $r(x) \neq 1$ stop since $r(x)$ is a factor of $f(x)$, otherwise

3. if $D = [f^2(x), g(x)f(x)]$ is singular we have $\gcd(x - g^2(x), f^2(x)) = h(x) \neq 1$ and $h(x)$ is a non-trivial factor of $f(x)$. In this case return $h(x)$ and stop. Otherwise do

4. compute $(p^i \pm 1)D$ by using above addition algorithm for $i = 0, \dots, n$

   $(p^i \pm 1)D$ gives either a factor of $f(x)$ or $[1, 0]$. If it gives a factor of $f(x)$, stop. Otherwise, assume $(p^i \pm 1)D = [1, 0]$ for some $i = 0, \dots, n$ and do

5. compute m and e such that $p^i \pm 1 = 2^e m$ with $(m, 2) = 1$ and $e > 0$.

6. compute $Q = mD$. If $Q = [1, 0]$ go to step 1 otherwise do

   repeat :

7. $Q_1 = 2^j Q$ for $j = 0, \dots e - 1$

8. $Q = Q_1 = [u, v]$

   until $v = 0$

9. if $u \neq x$ stop since in this case $u(x)$ is a multiple of non-trivial factor of $f(x)$, otherwise go to step 1

We need to do at most $2\widetilde{d}lgp$ steps of addition in the steps 4, 5 and 6 where $\widetilde{d}$ is maximum of the degrees of irreducible factors of $f(x)$. Performing addition of divisors requires approximately $O(n^2)$ bit operations. Overall the time taken to find a factor of $f(x)$ depends on cube of n and $logp$. Hence, Factorization Algorithm has almost the same asymptotic running time as the algorithm for Distinct Degree Factorization [9, 3.4.3] which is one of the necessary step for Cantor-Zassenhaus algorithm. On the other hand the well-known Berlekamp Algorithm has running time proportional to p [9, Algorithm 3.4.10]. We see from above discussion that the Algorithm 3 finds a non-trivial factor of $f(x)$ in a single trial with probability at least close to 7/8. However, Cantor-Zassenhaus algorithm finds a a non-trivial factor in a single trial with probability at most 1/2 and in the case of Berlekamp algorithm, the probability is less than 1/2 [9, Section 3]. We give examples in the last chapter to illustrate the above algorithms.

Chapter 5

Factorization of Polynomials mod p with Superelliptic Curves

5.1 Introduction

As we saw in the previous chapter, the idea of using hyperelliptic curves for factorization of polynomials and for computing square roots in finite fields provides very efficient methods for both problems. In this chapter, we investigate superelliptic curves for an answer to the natural question: How does the method for polynomial factorization work if one uses other curves instead of hyperelliptic curves?

The efficiency of the method in the previous chapter depends on the methods for computing in the Jacobian of a singular hyperelliptic curve. This is also true for the methods using superelliptic curves. Fortunately, there have been efficient methods obtained for computing in the divisor class group of a superelliptic curve.[1,11,12,13,14] The main idea of these methods come from computation algorithms in ideal class groups in number fields since the coordinate ring of the curve over the base field k is a Dedekind domain over $k[x]$ and its ideal class group is isomorphic to k-rational points of the Jacobian of the curve. From this, one can represent a divisor class in the Jacobian by the basis elements of the corresponding module over the ring $k[x]$.[1] The investigation of this representation shows that the n-torsion subgroup of the Jacobian of the curve $y^n = f(x)$ contains divisors classes whose representations consist of non-trivial factors of $f(x)$. This observation is one of the main ideas of the work in this chapter.

5.2 Superelliptic Curves

A superelliptic curve C over a field k is an algebraic curve with only one point at infinity and defined by an algebraic equation of the form $y^n = f(x)$ where $f(x)$ is a monic polynomial without repeated roots and n is coprime to $\deg(f(x))$ and $\mathrm{char} k$. The most efficient known algorithm takes $O(g^2)$ bit operations for an addition of divisors where g is the genus of the curve.[11]

The efficient methods for computations in the Jacobians of superelliptic curves are essentially analogous to the methods for computations in ideal class groups in number fields. The analogy is essentially based on the fact that the ideal class group of the coordinate ring $k[C]$ in the function field $k(C)$ is isomorphic to the k-rational points of the Jacobian of the curve.

The analogy of Hermite Normal Form in number fields gives that a divisor class in the Jacobian has a $k[x]$-module representative with a basis of the form $[a_{1,1}(x), \quad a_{2,2}(x)y + a_{2,1}(x), \quad \ldots, \quad a_{n,n}(x)y^{n-1} + \cdots + a_{n,1}(x)]$, where $a_{i,j}(x) \in k[x]$ and deg $a_{j,i}$ <deg $a_{i,i}$ for $1 \leq i < j \leq n$ with $a_{i+1,i+1}(x) | a_{i,i}(x)$ [11], [9]. This representation geometrically says that for any divisor class D there is a unique effective divisor E with $\deg E \leq g$ such that $D \sim E - m(\infty)$. From now on we represent a divisor class $D \in \mathrm{Jac}(C)$ as a $k[x]$-module (or an ideal of $k[C]$) with a basis of this kind.

Let $I = [a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \ldots, a_{n,n}(x)y^{n-1} + \ldots a_{n,1}(x)]$ be a $k(x)$-module representing a divisor class D. Then deg D=deg I =deg$(\prod_1^n a_{i,i}(x))$. We say that I is a reduced representative of D if it has the minimal degree in the divisor class D. Thus any divisor class D is uniquely represented by a reduced ideal I of the above form.[11, Proposition 4].

The following algorithm gives the reduced representative for a divisor class $D_1 + D_2$ (i.e. $I_1 \cdot I_2$)[11]

1. Find a representative for the multiplication $I = I_1 I_2$.

2. Find a representative for the inverse class I^{-1} of I.

3. Find an element α in I^{-1} such that it has a minimal norm in I^{-1}.

4. $I_3 = (\alpha)/I^{-1}$ is the reduced ideal in the class of I.

5.3 Polynomial Factorizations with Superelliptic Curves

Now we are going to use the same idea as described in chapter 3 to investigate the use of superelliptic curves for factorization of polynomials in finite fields. We first present the relation between the representative of a divisor class and the factors of a polynomial $f(x)$ for the superelliptic curve $C : y^n = f(x)$.

Lemma 5.3.1. *Let $C : y^n = f(x)$ be a superelliptic curve over a field k. Suppose d=deg $f(x)$ and $n \geq 3$. Then the curve C has geometric genus $g = (d-1)(n-1)/2$.*

Proof. The curve C is an n-fold cover of the projective line and there is only one point at infinity, denoted by ∞. Hence there are $d+1$ ramified points on C and the final result is from Hurwitz's Theorem. $\square$

Since the affine superelliptic curve C is non-singular, the coordinate ring $k[C]$ is a Dedekind domain which is also the integral closure of $k[x]$ in the function field $k[C]$. We also note that if ζ_n is a primitive n^{th} root of unity, the automorphisms $\sigma_n^i : (x, y) \to (x, \zeta_n^i y)$ of the curve C correspond to the elements of the Galois group G of $k(C)$ over $k(x)$.

Let $D = \sum_j m_j(P_j)$ be a divisor class in $\text{Jac}(C)$ represented by the $k[x]$-module $I = [a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \ldots, a_{n,n}(x)y^{n-1} + \cdots + a_{n,1}(x)]$. The first term $a_{1,1}(x)$ determines the first coordinates and multiplicities of the finite points P_j's in the support of D. That is, if $a_{1,1}(x) = \prod(x - c_j)^{m_j}$ then the point $P_j = (c_j, b_j)$ is in the support of D with multiplicity m_j. The elements $a_{i,i}(x)$ for $i > 1$ describe the first coordinates of the points P_j's whose ith Galois conjugates $P_j^{\sigma_n^i}$ are in the support of I. For example if $a_{2,2}(x) = \prod(x - c_r)^{m_r}$ then $P_r^{\sigma_n^2} = (c_r, \zeta_n^2 b_r)$ is in the support of D with multiplicity m_r.

Theorem 5.3.2. *Let $C : y^n = f(x)$ be a superelliptic curve over the field k and $I = [a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \ldots, a_{n,n}(x)y^{n-1} + \cdots + a_{n,1}(x)]$ be a $k[x]$-module corresponding to a divisor class D in Jac(C). If $a_{1,1}(x)$ is a factor of $f(x)$, then the order of the divisor class D divides n in Jac(C).*

Proof. Let $(x - a_i)$ be a factor of $f(x)$. Then $\text{div}(x - a_i) = n(P_i) - n(\infty)$ where (P_i) is the divisor class of the point $P_i = (a_i, 0)$. Thus the order of the divisor class of

$(P_i)-(\infty)$ divides n. Since $a_{1,1}$ is a factor of $f(x)$, each (P_k) in the sum $D = \sum_k (P_k)$ is of order dividing n. Hence D is of order dividing n. We note that each (P_k) has multiplicity one since $a_{1,1}(x)$ can not have a multiple root. □

Now consider the square-free reducible polynomial $f(x)$ over a finite field k. Let n be a prime number so that the curve defined by $C : y^n = f(x)$ is a superelliptic curve. Suppose #Jac$(C)(k)$=$n^e m$, (n, m)= 1, and $D \in$ Jac(C). If the order of D is divisible by n then $(n^i m)^{th}$ power of D must pass through an n-torsion of Jac(C) for some $i = 0, \ldots, e$. The above theorem says that the divisor classes represented by $I = [a_{1,1}(x), a_{2,2}(x)y + a_{2,1}(x), \ldots, a_{n,n}(x)y^{n-1} + \cdots + a_{n,1}(x)]$ with $a_{1,1}(x)$ a factor of $f(x)$ are of order n. Hence the $(n^i m)^{th}$ power of D may be in I and in this case the first coordinate of I gives us a non-trivial factor of $f(x)$.

The above discussion suggests the following factorization algorithm for a square-free reducible polynomial $f(x)$ over a finite field $\mathbb{F}_q$.

Factorization Algorithm:

1. Construct a superelliptic curve C by using the polynomial $f(x)$ so that a certain torsion subgroup of Jac(C), say S, contains divisor classes whose coordinates consist of factors of $f(x)$.

2. Find the order of Jac(C).

3. Select a random element D in Jac(C).

4. Determine if some power of D is in the subgroup S.

The idea of the above factorization algorithm is the same as the idea described in chapter 3. The efficiency of the constructed algorithm depends on:

1. The ratio of the number divisor classes in S whose coordinates consist of factors of $f(x)$ to the number of elements in S.
2. The efficiency of point counting in $\mathrm{Jac}(C)$.
3. The efficiency of finding an element in $\mathrm{Jac}(C)$.
4. The efficiency of computing in $\mathrm{Jac}(C)$.

Consider a square-free reducible polynomial $f(x)$ that we want to find the factors of. In most cases we consider either non-singular type $y^n = (x^i + a_0)f(x)$ for $i = 1$ or 2, or singular type $y^n = xf(x)^n$ to find a non-trivial factor of $f(x)$. If we consider the non-singular type, the first criterion about efficiency of the factorization algorithm suggests that n should be as small as possible, since from the previous theorem the subgroup that we are interested in is the subgroup of n-torsion of the Jacobian which is isomorphic to $\bigoplus_{i=1}^{2g} \mathbb{Z}/n\mathbb{Z}$ where g is the arithmetic genus of a curve. Considering the second and third criteria, the reasonable choice should be singular curves. The fourth one also recommends that the integer n should be as small as possible.

The factorization algorithm was investigated for $n = 2$ in chapter 3. Now based on the above discussion, the next choice should the superelliptic curves with $n = 3$ which are called superelliptic cubics.

5.4 Superelliptic Cubics

Let $C : y^n = f(x)$ be a superelliptic curve over a field k. Since the group of k-rational points of $\mathrm{Jac}(C)$ is isomorphic to the group of isomorphism classes invertible $\mathcal{O}_C(C) = k[C]$-modules, we first describe the $\mathcal{O}_C(C)$-module representation of a divisor class in the Jacobian of a superelliptic cubic.

Proposition 5.4.1. *Let $C : y^n = f(x)$ be a superelliptic curve over a field k. If the support of a divisor class D does not contain a pair of Galois conjugate points, then D can be represented by an ideal of the form $(u, y - v)$ satisfying*

1. *$u(x)$ is a monic polynomial*

2. *deg(v) <deg(u)*

3. *$u(x)$ divides $f(x) - v(x)^n$*

Proof. The idea comes from the Mumford Representation for the divisors in the Jacobians of hyperelliptic curves[17]. Let $D = \sum n_i(P_i)$ with $P_i = (a_i, b_i)$ on C. Consider the polynomial $u(x) = \prod_i (x - a_i)^{n_i}$. Then if we solve the congruences $w(x)^n \cong f(x) \pmod{(x - a_i)^{n_i}}$ for all i with $w(a_i) = b_i$ as in [24, Section 13.2] and combine them by using Chinese Remainder Theorem, we get the polynomial $v(x)$ satisfying the above conditions. □

Remark 5.4.2. Consider the polynomial $(x-a)$ and its divisor class div($(x-a)$)$=$ $P + P^{\sigma_n} + \cdots + P^{\sigma_n^{n-1}} - n(\infty)$ where $\sigma_n^i \in \mathrm{Gal}(k(C)/k(x))$. We can add a suitable multiple of such a divisor class to any random divisor class. Therefore we can say that any divisor class $D = \sum m_i(P_i)$ has a representative $\widetilde{D}$ having at most $n-1$ conjugate pairs of a point,

The above theorem essentially says that a nice compact representation for a divisor class is possible if the divisor class has a representative without a pair of Galois conjugate points. This nice condition is satisfied by all divisor classes if $n = 2$, i.e. if the curve is hyperelliptic. As for $n = 3$ the majority of divisor classes has a representative without a pair of conjugate points [3]. As n gets larger, the fraction of divisor classes having nice representatives decreases. This compact representation provides some efficiency for computations in Jacobians.

Now consider the coordinate ring of the superelliptic cubic $C : y^3 = f(x)$ over the field k. We use $k[x]$-module representations of the ideals of $k[C]$ for computations in $\mathrm{Jac}(C)$, as in [11] or [2]. Note that any ideal class has a unique minimal degree $k[x]$-module representative of the form, called canonical form, $I = [s, s'(y+u), y^2 + wy + v]$ satisfying

1. $s'|s$
2. $u^3 \equiv -f(\bmod\ s/s')$
3. $v \equiv w^2 \pmod{s'}$
4. $v - uw + u^2 \equiv 0 \pmod{s/s'}$
5. $uv - uw^2 \equiv f - vw \pmod{s}$. [2, Section 4] or [19]

The canonical basis for an ideal is called *minimal canonical basis*, if it also satisfies

1. s and s' are monic,
2. deg$(s'u)$ <deg(s),
3. deg(v) <deg(s) and deg(w) <deg(s').

The degree of a divisor is deg(ss'). There exits a unique minimal canonical basis for any divisor class. For more details see [2, Section 4]

Let I_1 and I_2 be reduced representatives of divisor classes in Jac(C). The first step for the addition algorithm is to find a representative for $I_1 \cdot I_2$. This can be done by using the following algorithm[2, Section 7]. Note that there is a misprint in step 9 in Bauer's paper [2].

Algorithm 1: Input $I_i = [s_i, s_i'(y+u), y^2 + w_i y + v_i]$ for $i = 1, 2$

1. Compute d, r_1 such that d=gcd$(s_1/s_1', s_2/s_2')$=$r_1 s_1/s_1' + r_2 s_2/s_2'$.
2. d_1=gcd$(d, u_1 - u_2)$/gcd(d, f)
3. $S = s_1 s_2 d_1/d, \qquad S' = s_1' s_2' d/d_1 \qquad$ and $u = u_1 - (u_1 - u_2)(r_1 s_1/(s_1' d))$
4. Compute d_2=gcd$(d_1, 3u^2)$=$3r_2u^2 + r_4 d_1$
5. $U' = u - r_3(u^3 + f)/d_2$
6. $U \equiv U'($mod $S/S')$ such that deg$U$ <degS/S'
7. Compute 1=gcd$(s_1, s_1's_2', s_1'(u_1 + w_2), s_2, s_2'(u_2 + w_1), v_1 + v_2 + w_1 w_2) = a_1 s_1 + a_2 s_1' s_2' + a_3 s_1'(u_1 + w_2) + a_4 s_2 + a_5 s_2'(u_2 + w_1) + a_6(v_1 + v_2 + w_1 w_2)$ for some $a_1, \ldots, a_6$ in k

8. $V' = a_1 s_1 v_2 + a_2 s_1' s_2' u_1 u_2 + a_3 s_1'(u_1 v_2 + f) + a_4 s_2 v_1 + a_5 s_2'(u_2 v_1 + f) + a_6(v_1 v_2 + w_1 f + w_2 f)$

9. $W' = a_1 s_1 w_2 + a_2 s_1' s_2'(u_1 + u_2) + a_3 s_1'(u_1 w_2 + v_2) + a_4 s_2 w_1 + a_5 s_2'(u_2 w_1 + v_1) + a_6(w_1 v_2 + v_1 w_2 + f)$

10. $W = W' + qS'$ such that deg$W <$degS'

11. $V \equiv V' + qS'U(\text{mod } S)$ such that deg$V <$deg$S$

Output: $I = I_1 \cdot I_2 = [S, S'(U + y), y^2 + Wy + V]$

As we mentioned above, in most cases there would be no Galois conjugate pairs in the support of a divisor class, therefore $S' = 1$ in most cases. If this is the case, we change the steps after Step 6 in Algorithm 1 to:

(7) $W = 0$ and $V \equiv -U^2 \pmod S$

Output: $I = I_1 \cdot I_2 = [S, S'(U + y), y^2 + Wy + V]$

There are also some algorithms for addition operation in Jac(C) which only use the compact representations, $(u, y - v)$, of divisors classes in Jac(C). Although asymptotically each addition operation has the same running time for hyperelliptic curves and superelliptic curves, there is a big difference in real time implementation. The difference comes from especially the reduction operation after ideal multiplication.

5.5 Factorization with Superelliptic Cubics

Now we compare hyperelliptic and superelliptic cubics in terms of polynomial factorizations in finite fields. From now on we assume $k = \mathbb{F}_q$ is a finite field with $q = p^e$ elements.

Let $f(x)$ be a square-free polynomial in $k[x]$ such that gcd(deg($f(x)$,3)=1, and $C : y^3 = f(x)$ be a superelliptic curve over k. If $f_i(x)$ is a non-trivial factor of $f(x)$ then by Theorem 5.3.2 the divisor class $D_i = [f_i(x), 1, 0, 0, 0]$ is of order 3 in Jac(C). Note that a divisor corresponding to a $k[x]$-module of the form $[f_i(x), 1, 0, 0, 0]$ is a reduced divisor. Now, if a divisor class $\widetilde{D}$ is of order a multiple of 3, a certain power of it might be in the divisor class D_i. Hence a factor of $f(x)$ would be found. Unlike divisors in the Jacobians of hyperelliptic curves, some divisor classes of order 3 do not contain a factor of $f(x)$.

Example: Let $f(x) = x^4 + 6x^2 + 12$ and $C : y^3 = f(x)$ be a superelliptic curve over $\mathbb{F}_{13}$. The divisor class $D = 2(P) - 2(\infty)$ where $P = (0, 4)$ is represented by the ideal $I = [x^2, y + 9, y^2 + 10]$. By using above addition algorithm we can show that $D + D = [x^2 + 6, y + 9, y^2 + 10] = -D$, hence D is of order 3 in Jac(C) and the corresponding ideal of it does not have any factor of $f(x)$.

The above discussion shows that based on the first condition about efficiency of the factorization algorithm, hyperelliptic curves have advantages over superelliptic cubics. This is because all ideal representations of 2-torsion points of the Jacobian of a hyperelliptic curve are in the same format. In terms of the second and the third criteria in section 2, there is no difference between using a hyperelliptic curve or superelliptic cubics if we are allowed to use singular curves for both cases. Hence the remaining part is to compare hyperelliptic curves and superelliptic cubics in terms of efficiency of computations in Jacobians. As we noted above, the asymptotic running time of computation in Jacobians is the same for hyperelliptic and superelliptic curves but real running time is not the same. To compare these two kinds of curves in terms of computations in Jacobians we first state a factorization algorithm with non-singular superelliptic curves.

Algorithm 2: Input: $f(x)$, which is a square-free reducible polynomial in $k[x]$.

1. Pick a random number $a \in k$
2. (Depends on $\deg(f(x))$) construct either the curve $C : y^n = (x + f(a)^{n-1} - a)f(x)$ or the curve $C : y^n = (x^2 + f(a)^{n-1} - a)f(x)$ where n is a prime number
3. Find the order of Jac(C)=$n^e m$, $(n, m) = 1$
4. Find $\widetilde{D} = mD$ where $D = [x - a, (y - f(a)), y^2 - f(a)^2, \dots, y^{n-1} - f(a)^{n-1}]$
5. if $\widetilde{D} = [1, y, y^2, \dots y^{n-1}]$ go to step 1 otherwise compute $P_i = n^i \widetilde{D}$ for $i = 0, \dots, e$
6. If none of $a_{1,1}(x)$ divides $f(x)$ where $P_i = [a_{1,1}(x), \dots, a_{n,n}y^{n-1} + \dots + a_{n,1}]$ go to step 1 otherwise $a_{1,1}(x)$ is a factor of $f(x)$.

We now give an example to illustrate the above algorithm by using superelliptic cubics.

Example 5.5.1. Consider the polynomial $f(x) = x^4 + x^3 + 4x^2 + 2x + 4$ over the field $\mathbb{F}_5$.

Step 1: We start with $a = 0$ and construct the curve $C : y^3 = (x + f(0)^2)f(x) = (x + 16)f(x) = x^5 + 2x^4 + x^2 + x + 4$

Step 2: We compute $\text{Jac}(C) = 360 = 3^2(40)$

Step 3: $D = [x, y - 4, y^2 - 4^2]$ and $\widetilde{D} = 40D = [x^4, y + 2x^3 + 4x^2 + 3x + 1, y^2 + 4 + 3x^2 + 2x^3 + 4x]$ which is not the identity element of $\text{Jac}(C)$, hence the order of D is divisible by 3

Step 4: $\gcd(x^4, f(x)) = 1$, then we compute $3\widetilde{D} = [x^3 + x^2 + 2x + 2, y, y^2]$ and $\gcd(x^3 + x^2 + 2x + 2, f(x)) = x^2 + 2$

In order to compare the running time of the algorithms with hyperelliptic curves, we use exactly the same method with $n = 2$.

Step 1: We again start with $a = 0$ and construct the curve $H : y^2 = (x + f(0))f(x) = (x + 4)f(x)$

Step 2: We compute $\text{Jac}(H)$=20=$2^2 5$

Step 3: $D = [x, y - 4]$ and $\widetilde{D} = 5D = [x^2 + x + 2, 0]$ and $x^2 + x + 2$ must be a factor of $f(x)$.

The real time for implementation of Algorithm 2 with superelliptic curves shows that the algorithm 2 works much slower than the same algorithm with hyperelliptic curves because of the running time of addition operation for superelliptic curves. The experiments that we conducted over finite fields with characteristic very small prime numbers p shows that the algorithm with hyperelliptic curves finds a factor of a polynomial at least 40 times faster than the algorithm with superelliptic cubics does.

The most efficient factorization algorithm in chapter 3 uses singular hyperelliptic curves. A similar idea can be extended for using singular superelliptic curves. In this case we construct a singular superelliptic curve $C : y^3 = xf(x)^3$. In this way, we cancel the step that we count the number of points on Jacobian since $\mathrm{Jac}(C)$ is of isomorphic to $\bigoplus_i^g \mathbb{G}_m$ where g is the arithmetic genus of C . Unfortunately, the algorithm will also be less efficient because of computations in the Jacobians of superelliptic curves.

Chapter 6

Examples

In this chapter we give examples to illustrate the algorithms described in former chapter.

6.1 Examples: Computing Square Roots mod p

Example 6.1.1. Let $p = 1049219$ and $a = 123451$ in $k = \mathbb{F}_p$. Since $p \equiv 3 \pmod 4$ we use *Algorithm 2 in section 3.3.1.*

1. Consider the singular hyperelliptic curve $H : y^2 = x(x^2 - 123451)^2$ over k. Take $D = [(x^2 - 123451)^2, (x^2 - 123451)]$ in $\mathrm{Jac}(H)$.
2. To find $\sqrt{a}$ mod p it is enough to compute $(p-1)D$ in $\mathrm{Jac}(H)$.
3. $(p-1)D = [x^2 + 930173x + 697558, 110955x + 972129]$
4. Now compute $\gcd(x^2 + 930173x + 697558, x^2 - 123451) = 1$.

Hence we conclude that $a = 123451$ is not a square mod 1049219.

Example 6.1.2. Let $p = 31476587$ and $a = 5711954$. Once again $p \equiv 3 \pmod 4$ and we use *Algorithm 2 in section 3.3.1*

1. Let $H : y^2 = x(x^2 - 5711954)^2$ and $D = [(x^2 - 5711954)^2, (x^2 - 5711954)]$. Then

2. $(p - 1)D = [x^2 + 15616214x + 5711954, 11540220 + 2096153x]$

3. $\gcd(5788728x + 21357467, x^2 - 5171954) = x + 7808107$ which means 7808107 is $\sqrt{5711954} \bmod 31476587$

Actually it is not necessary to use gcd if we know $a = 5711954$ is a square mod p since we know that $(p-1)D = [(x \pm \sqrt{a})^2, t(x \pm \sqrt{a})]$ for some t then we have $11540220 + 2096153x = t(x + \sqrt{a})$ this implies $11540220/2096153 \equiv 7808107 \pmod{31476587}$ is a square root of a modulo p.

Example 6.1.3. Let $p = 35019169$ and $a = 610623$. Since $p \equiv 1 \pmod 8$ we use *Algorithm 1 in section 3.2.1.* Let E be the elliptic curve defined by $y^2 = x(x + 610623)^2$

1. Pick a random number b and consider the point $P = (b^2, b(b + 1)) = [(x + 610623)^2, b(x + 610623)]$ on the curve E. Let's pick $b = 1$ so $P = (1, 610624)$

2. Find e and odd number m such that $p - 1 = 35019168 = 2^e m$ in this case $e = 5$ and $m = 1094349$

3. Compute $Q = mP = (23855786, 13003707)$. That means the order of P is even

4. Compute $Q_1 = 2Q = (13947345, 32162710)$. The first coordinate is not a so set $Q = Q_1$

5. Compute $Q_1 = 2Q = (610623, 20530334) = (a, 2a\sqrt{a})$

6. Compute $20530334/(2 \cdot 610623) = 16014346$ which is equal to $\sqrt{610623}$

Example 6.1.4. Let p and a be the same as above. In this case we are going to use *Algorithm 3 in section 3.3.2* to compute $\sqrt{a} \bmod p$.

1. Pick a random b and we pick $b = 1$.

2. Let $H : y^2 = x(x^2 - 2bx + b^2 - a)^2 = x(x^2 - 2x - 610622)^2$ be a singular hyperelliptic curve. Let $D = [(x^2 - 2x - 610622)^2, x^2 - 2x - 610622] \in \text{Jac}(H)$

3. Compute $(p-1)D$. We get $(p-1)D = [x^2 + 35019167x + 1, 17204273x + 17204273]$ which is not of the form $[(x - (b \pm \sqrt{a})^2, i(x - (b \pm \sqrt{a})]$. Hence we have $(p+1)D = [1, 0]$.

4. Compute e and odd number m such that $p + 1 = 35019168 = 2^e m$. In this case $e = 1$ and $m = 17509585$

5. Compute $mD = [x, 0]$. For $b = 1$ we did not get $\sqrt{a} \pmod{p}$. We repeat this operation by replacing $b = 2$.

6. In this case we $(p-1)D = [x^2 + 2990473x + 29648842, 21384677x + 4346851] = [(19004821 + x)^2, 21384677(19004821 + x)]$

7. $x + 19004821 = x - (b \pm \sqrt{a})$ hence $\sqrt{a} = 19004823 \equiv -16014346 \bmod (p)$

6.2 Polynomial Factorization mod p

The following example illustrates the factorization method described in chapter 4.

Example 6.2.1. Let $f(x) = 3655 + 3827x + 3224x^2 + 6323x^3 + 3085x^4 + 3702x^5 + 411x^6 + 1234x^7 + 191x^8 + 104x^9 + 26x^{10} + 23x^{11} + x^{12}$ and let the field $k = \mathbb{F}_{571}$. Now consider the singular hyperelliptic curve $H : y^2 = xf^2$. We use the addition algorithm described in chapter 4 for divisor classes in the Jacobian of a singular hyperelliptic curve.

Step 1: We select $g(x) = 1$ hence the divisor class $D = [f^2, f]$.

Step 2: Compute $(571^i \pm 1)D$ until we get a factor of $f(x)$ or $[1, 0]$ for $i = 1, \ldots, 12$.

$(571-1)D = [f^2, (534x^{11}+186x^{10}+355x^9+412x^8+323x^7+559x^6+264x^5+181x^4+496x^3+534x^2+535x+384)f]$

$(571+1)D = [f^2, (334x^{11}+509x^{10}+294x^9+431x^8+365x^7+508x^6+378x^5+528x^4+374x^3+372x^2+500x+562)f]$

$(571^2-1)D = [f^2, (242x^{11}+42x^{10}+116x^9+259x^8+283x^7+411x^6+238x^5+570x^4+317x^3+556x^2+542x+184)f]$

$(571^2+1)D = [f^2, (142x^{11}+126x^{10}+360x^9+342x^8+500x^7+35x^6+541x^5+460x^4+92x^3+242x^2+492x+418)f]$

$(571^3 - 1)D = [f^2, (x^{11} + 287x^{10} + 290x^9 + 71x^8 + 350x^7 + 46x^6 + 76x^5 + 234x^4 + 539x^3 + 74x^2 + 314x + 388)f]$

$(571^3 + 1)D = [f^2, (200x^{11} + 23x^{10} + 91x^9 + 439x^8 + 432x^7 + 346x^6 + 191x^5 + 328x^4 + 336x^3 + 235x^2 + 463x + 491)f]$

$(571^4 - 1)D = [(x^4 + 22x^3 + 11x + 17)^2, (247x^3 + 39x^2 + 48x + 3)(x^4 + 22x^3 + 11x + 17)]$

The last part of Step 2 shows that $f_1(x) = x^4 + 22x^3 + 11x + 17$ is a factor of $f(x)$. Now we apply the same method for the remaining factor $h(x) = x^8 + x^7 + 4x^6 + 5x^5 + 53x^4 + 7x^3 + 134x^2 + 86x + 215$ of $f(x)$. We first define the singular hyperelliptic curve $C : y^2 = xh^2$.

Step 3: We select $\widetilde{D} = [h^2, h]$ by choosing $g(x) = 1$ again.

Step 4: Compute $(571^i \pm 1)\widetilde{D}$ until we get a factor of $h(x)$ or $[1, 0]$ for $i = 1, \ldots, 8$. These computations yield that $(571^4 - 1)\widetilde{D} = [1, 0]$. Now we check if the order of D is even by looking 6643920855^{th} power of $\widetilde{D}$, since $571^4 - 1 = 2^4(6643920855)$.

Step 5: $D' = 6643920855\widetilde{D} = [x^8 + 209x^7 + 508x^6 + 83x^5 + 211x^4 + 102x^3 + 101x^2 + 5x + 81, 428 + 335x + 46x^6 + 322x^2 + 8x^4 + 77x^3 + 103x^7 + 309x^5]$.

This shows that the order of D is even.

Step 6: Compute $2^j D'$ until the result is 2-torsion of $\text{Jac}(C)$ for $j = 1, \ldots 4$.

$2D' = [x^8 + 187x^7 + 504x^6 + 445x^5 + 524x^4 + 405x^3 + 129x^2 + 438x + 418, \quad 50 + 547x + 386x^6 + 346x^2 + 190x^4 + 453x^3 + 272x^7 + 194x^5]$

$4D' = [x^8 + 493x^7 + 393x^6 + 202x^5 + 531x^4 + 216x^3 + 85x^2 + 521x + 25, \quad 294 + 297x + 355x^2 + 427x^4 + 113x^7 + 121x^3 + 311x^6 + 344x^5]$

$8D' = [x^8 + 2x^7 + 3x^6 + 2x^5 + 87x^4 + 86x^3 + 86x^2 + 136, \quad 0]$

then $\gcd(x^8+2x^7+3x^6+2x^5+87x^4+86x^3+86x^2+136, h(x)) = x^4+x^3+x^2+43$ is a factor of $f(x)$. Therefore the factors are $f_1 = x^4+x^3+x^2+43$, $f_2 = x^4+3x^2+2x+5$ and $f_3 = x^4 + 22x^3 + 11x + 17$.

These computations show that $\text{Jac}(H) = \mathbb{G}_1 \oplus \mathbb{G}_2 \oplus \mathbb{G}_3$ where each $\mathbb{G}_i$ is a cyclic group. The elements of each $\mathbb{G}_i$ is represented by a pair of the form $[f_i(x)^2, t_i(x) f_i(x)]$ where $t_i(x)$ is a polynomial of degree less than 4. We see in step 2 that $\mathbb{G}_1$ and $\mathbb{G}_2$ are of order $571^4 - 1$ and $\mathbb{G}_3$ is of order $571^4 + 1$ and this explains why we get a factor in Step 2.

Bibliography

[1] G.W. Anderson, Abeliants and their application to elementary construction of Jacobians, Advances in Mathematics 172, 169-205, 2002.

[2] M.L. Bauer, The Arithmetic of Certain Cubic Function Fields, Math.Comp, (73)(2003), 387-413.

[3] A. Basiri, A. Enge, J-H. Faugere, N. Gurel, The Arithmetic of Jacobian Groups of Superelliptic Curves, Math.Comp, (74)2004, 389-410.

[4] E. Berlekamp, Factoring polynomials over large finite fields, Math. Comp. 24 (1970), 713-735.

[5] S. Bosch, W. Lutkebohmert, M. Raynaud, Neron Models, Springer-Verlag, 1990.

[6] D. G. Cantor, Computing in the Jacobian of a hyperelliptic curve, Math. Comp. 48 (1987), 95-101.

[7] D. Cantor and H. Zassenhaus, A new algorithm for factoring polynomials over finite fields, Math. Comp., 36 (1981), 587-592.

[8] H. Cohen, G. Frey, Handbook of Elliptic and Hyperelliptic Curve Cryptography, Chapman & Hall/CRC 2005.

[9] H. Cohen, A Course in Computational Algebraic Number Theory, Springer-Verlag, 2000.

[10] D. Eisenbud, Commutative Algebra with a View Toward Algebraic Geometry. Springer-Verlag, 2004

[11] S.D. Galbraith, S.M. Paulus, N.P. Smart, Arithmetic on Superelliptic Curves, Math.Comp, (71)2000, 393-405.

[12] F. Hess, Zur Divisorenklassengruppenberechnung in globalen Funktionenkorpern, Ph.D. thesis, Technische Universitat Berlin, 1999,

[13] K. Khuri-Makdisi, Linear algebra algorithms for divisors on an algebraic curve. Math. Comp. 73 (2004), no. 245, 333–357

[14] K. Khuri-Makdisi, Asymptotically fast group operations on Jacobians of general curves. Math. Comp. 76 (2007), no. 260, 2213–2239

[15] D. R. Kohel, Constructive and destructive facets of torus-based cryptography, pre-print, available at http://echidna.maths.usyd.edu.au/ kohel/index.html

[16] Q. Liu, Algebraic Geometry and Arithmetic Curves, Oxford Science Publications, 2002.

[17] D. Mumford, Tata Lectures on Theta II, Birkhauser, 1982.

[18] M. Rosenlicht, Equivalence relations on algebraic curves, Ann. of Math. 56, 169-191 (1952).

[19] R. Scheidler, Ideal arithmetic and infrastructure in purely cubic function fields, J. Theor. Nombres Bordeaux 13 (2002), 609-631.

[20] R. Schoof, Elliptic curves over finite fields and the computation of square roots mod p, Math. Comp. 43(1985), 483-494.

[21] J. P. Serre, Algebraic Groups and Class Fields,Springer-Verlag, 1997.

[22] D. Shanks, Class number, a theory of factorization, and genera, Proc. Symp in Pure Maths. 20, AMS, Providence, R.I., 1971, pp. 415-440.

[23] J. H. Silverman, Advanced Topics in the Arithmetic of Elliptic Curves, Springer-Verlag, 1992.

[24] L. C. Washington, Elliptic Curves: Number Theory and Cryptography, 2nd edition. Chapman & Hall/CRC 2008