GROUP LAW ALGORITHMS FOR JACOBIAN VARIETIES OF CURVES OVER FINITE FIELDS

RAN COHEN

ABSTRACT. This paper reviews the group law algorithms of nonsingular C_{ab} curves, and provides implementation results comparing it to an algorithm for computing in the generalized Jacobian of C_A curves.

1. INTRODUCTION

The success of Jacobian groups of (nonsingular) elliptic and hyperelliptic curves in cryptography has drawn a lot of interest over the recent years. One of the main areas of research is to find suitable Jacobian groups of other families of curves, and implement their group operation efficiently. The first part of this work is a survey of the group law algorithms of nonsingular C_{ab} curves. The second part is a review of the algorithm, proposed recently by Arita, Miura and Sekiguchi, for generalized Jacobian of a special family of singular curves, so called C_A curves. Implementation results provide a comparison of the running time between C_{ab} curves and C_A curves.

In this paper we follow the notations of [Sil86]. Let $\Bbbk = \mathbb{F}_q$, where $q = p^t$ for some prime p, and denote by $\overline{\Bbbk}$ its algebraic closure. Let $C_0/\Bbbk$ be an absolutely irreducible projective curve. We fix a $\Bbbk$-rational point $P_\infty \in C_0$, and let C be the curve obtained by desingularizing only the point P_∞. We assume there is only one point lying above P_∞ in C which is also denoted P_∞. In addition, we assume that $C_a = C \backslash \{P_\infty\}$ is a nonsingular affine curve. Let $R = \Bbbk[C_a]$ be its coordinate ring, and $\Bbbk(C)$ its field of rational functions. Let g be the genus of the curve C.

The Jacobian variety $\mathrm{Jac}(C)$ is isomorphic as a group to the degree zero subgroup of the Picard group $\mathrm{Pic}^0(C) := \mathrm{Div}^0(C)/\mathrm{PDiv}(C)$. We focus on $\mathrm{Pic}^0_\Bbbk(C)$, the invariant subgroup w.r.t. $\mathrm{Gal}(\overline{\Bbbk}/\Bbbk)$. Recall that a divisor of the form $E - nP_\infty$, where $E \geq 0$ is affine and of degree n, is called a *semi-reduced divisor* with *weight* n. Using Riemann-Roch Theorem one can prove that every divisor class $[D] \in \mathrm{Pic}^0_\Bbbk(C)$ has a unique representative $D \sim E - mP_\infty$, with minimal weight m. The divisor $E - mP_\infty$ is called a *reduced divisor*.

The coordinate ring R is obviously Noetherian and of Krull dimension 1. For curves the notions of normality and nonsingularity are equivalent, therefore C_a is a normal curve. It follows that R is a Dedekind domain, and the ideal class group $\mathrm{IdCl}(R)$ is well defined. In this case we have a natural isomorphism between $\mathrm{Pic}_\Bbbk(C_a)$ and $\mathrm{IdCl}(R)$. In addition, P_∞ is the single point at infinity, and so we have an isomorphism between $\mathrm{Pic}^0_\Bbbk(C)$ and $\mathrm{Pic}_\Bbbk(C_a)$, by $\sum n_P P - (\sum n_P) P_\infty \leftrightarrow \sum n_P P$. For conclusion, we get the following sequence of isomorphisms of groups:

$$\mathrm{Jac}_\Bbbk(C) \simeq \mathrm{Pic}^0_\Bbbk(C) \simeq \mathrm{Pic}_\Bbbk(C_a) \simeq \mathrm{IdCl}(R).$$

Date: December 15, 2007.

Key words and phrases. generalized Jacobian, hyperelliptic curves, cryptography.

2. C_{ab} Curves

Let $C/\Bbbk$ be a plane irreducible projective curve, and let $P \in C$. We define

$$\mathcal{L}(\infty P) := \bigcup_{n \geq 0} \mathcal{L}(nP).$$

Define $M(P) = \left\{ -\operatorname{ord}_P(f) : f \in \mathcal{L}(\infty P) \backslash \{0\} \right\}$. If $a, b \in M(P)$ there exist functions $f, h \in \mathcal{L}(\infty P)$ such that $-\operatorname{ord}_P(f) = a$ and $-\operatorname{ord}_P(h) = b$. Clearly $fh \in \mathcal{L}(\infty P)$, and $-\operatorname{ord}_P(fh) = a + b$, so $M(P)$ is a unitary semigroup w.r.t. addition.

Definition 2.0.1 (C_{ab} curve, [Ari99])**.** If the semigroup $M(P)$ is generated by two relatively prime positive integers a and b, then the pair (C, P) is called a C_{ab} *curve*.

Let (C, P) be a C_{ab} curve. Then by definition there are functions $x, y \in \mathcal{L}(\infty P)$ with poles of order a and b respectively. Using these two functions we obtain the affine model of the C_{ab} curve $F(x, y) = \sum_{ai+bj \leq ab} c_{ij} x^i y^j = 0$, where $0 \leq i \leq b$, $0 \leq j \leq a$, $c_{ij} \in \Bbbk$, $c_{b0} \neq 0$ and $c_{0a} \neq 0$. This affine model of C is called the *Miura canonical form*. We assume that C_{ab} curves satisfy the conditions in the introduction, i.e. nonsingular in the affine plane, and P is the only point at infinity, denoted P_∞. The genus of a C_{ab} curve is $g = \frac{1}{2}(a-1)(b-1)$.

Definition 2.0.2 (C_{ab} order, [Ari99])**.** Let $\alpha = (\alpha_1, \alpha_2), \beta = (\beta_1, \beta_2) \in \mathbb{N}^2$. We say that $\alpha >_{ab} \beta$, if one of the following holds:

(1) $a\alpha_1 + b\alpha_2 > a\beta_1 + b\beta_2$.
(2) $a\alpha_1 + b\alpha_2 = a\beta_1 + b\beta_2$, and $\alpha_1 < \beta_1$.

We can order the monomials $x^{\alpha_1} y^{\alpha_2} \in \Bbbk(C)$ by their pole order at P_∞, by setting $-\operatorname{ord}_{P_\infty}(x^{\alpha_1} y^{\alpha_2}) = a\alpha_1 + b\alpha_2$, and when two monomials have the same pole order at infinity, the monomial with the larger degree of x is smaller.

2.1. Classification of algorithms.

Various algorithms were proposed for Jacobians of C_{ab} curves, with special care for curves appealing for cryptography such as C_{25} and C_{34} curves. The algorithms consist of two stages: composition and reduction, and can be classified to the following types:

The first type is based on *hyperplane intersection*. The underlying idea is to construct a hyperplane interpolating the affine points in the support of two reduced divisors. The intersection of this hyperplane with the curve forms a divisor in the opposite class. Inverting this divisor yields the reduced divisor.

The second type is based on Cantor's algorithm. This approach is inspired from Gauß' algorithm for quadratic forms, which was utilized by E. Artin to the case of hyperelliptic fields, quadratic extensions of $\Bbbk(x)$. Cantor used this approach for computing in Jacobians of hyperelliptic curves.

The third type is based on Gröbner basis manipulation and includes two subfamilies: algorithms based on lexicographic order and algorithms based on C_{ab} order. Algorithms in the first subfamily use an LLL-like algorithm in order to reduce a divisor. This algorithm finds a reduced basis for a lattice in a function field, allowing us to compute the minimal element in the lattice based on a specific metric. Algorithms in the second subfamily compute the minimal element in the reduced Gröbner basis (with respect to C_{ab} order) of the ideal corresponding to the divisor. The divisor is then reduced using this minimal element.

In the remaining part of this section we review special families of C_{ab} curves, and the algorithms proposed for each family based on chronological order.

2.2. **Elliptic Curves.** The first example of C_{ab} curves are C_{23} curves, elliptic curves. By definition, elliptic curves are nonsingular curves of genus 1 along with a fixed base point. In this case the Miura canonical form is mostly known as the Weierstraß form, $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$, where the fixed point P_∞ corresponds to $(0, 1, 0)$.

The set of rational points on an elliptic curve has a natural composition, the *chord and tangent* law. This algorithm obviously belongs to the first type, hyperplane intersection.

Based on the chord and tangent law, one can derive explicit formulae valid over any field. These formulae can be simplified based on the characteristic of the field. The efficiency of the computation can be further improved based on the coordinates system: affine coordinates, projective coordinates, Jacobian coordinates, Chudnovski-Jacobian coordinates, modified Jacobian coordinates, or mixed coordinates (see [CMO98]).

2.3. **Hyperelliptic Curves.** A *hyperelliptic curve* $C/\Bbbk$ is a projective curve of genus $g \geq 1$ that admits a nonsingular affine model of the form $y^2 + H(x)y = F(x)$, where $H, F \in \Bbbk[x]$, F is monic of degree $2g + 1$, and $\deg(H) \leq g$. Clearly, hyperelliptic curves are C_{2b} curves. If $\mathrm{char}(\Bbbk) = 2$ then we have $H(x) \neq 0$, and if $\mathrm{char}(\Bbbk) \neq 2$, C can be represented in the form $y^2 = F(x)$. The order of the Jacobian of a hyperelliptic curve is approximately q^g, so we can obtain a similar group as of an elliptic curve while working over a field with $\sqrt[g]{q}$ elements.

In ([Mum84]) Mumford showed that every semi-reduced divisor on a hyperelliptic curve can be represented as the gcd of two principal divisors. Using Mumford's representation one can represent a divisor over $\Bbbk$ even if its support is contained in some extension of $\Bbbk$. Let $D = \sum n_P P - \big(\sum n_P\big)P_\infty$ be a semi-reduced divisor, and set $U(x) = \prod(x - x_P)^{n_P} \in \Bbbk[x]$, then there is a unique polynomial $V(x) \in \Bbbk[x]$ satisfying: $\deg(V) < \deg(U)$, $V(x_P) = y_P$ for all P for which $n_P \neq 0$, and $U|(V^2 + VH - F)$. It follows that $D = \gcd\big(\mathrm{div}(U), \mathrm{div}(y - V)\big)$. We simplify this notation to $\mathrm{div}(U, y - V)$.

In [Gau00] Gaudry used a variant of index calculus for the Jacobian group of hyperelliptic curves of genus g defined over $\mathbb{F}_q$, and managed to compute the DLP with complexity $O(q^2)$. Note that the group size is approximately q^g, and so Pollard's Rho attack computes the DLP with complexity $O(q^{g/2})$. Thus, Gaudry's attack is faster than Pollard's Rho when the genus is greater than 4.

2.3.1. *Cantor's algorithm.* In 1987 Cantor utilized Mumford's representation to propose an algorithm for addition in the Jacobian of hyperelliptic curves over fields of odd characteristic (see [Can87]). In 1989 Koblitz generalized Cantor's algorithm to arbitrary characteristic (see [Kob89]). This algorithm consists of two steps: composition and reduction. The reduction algorithm is based on the classical method due to Gauß. Nevertheless, there are other reduction algorithms that are faster asymptotically, i.e. as the genus grows larger.

In 2000 Nagao improved Cantor's algorithm over odd characteristic, based on the following ideas: (see [Nag00])

- Dividing polynomials without inversion in the base field.
- Computing gcd of polynomials using only one inversion in the base field.
- Ignoring superfluous operations during the algorithm.
- Representing the Jacobian's elements differently.

2.3.2. *Harley's algorithm.* In 2000 Harley proposed a generalization of the chord and tangent law to the case of hyperelliptic curves of genus 2 over odd characteristic ([Har00a, Har00b]). This algorithm is based on hyperplane intersection. Given two reduced divisors D_1 and D_2, after the composition step we have a semi-reduced divisor R of weight at most $2g$. If $\text{weight}(R) \leq g$ then R is already reduced, otherwise, for genus 2, $\text{weight}(R)$ can be either 3 or 4.

If $\text{weight}(R) = 3$, then $R = P_1 + P_2 + P_3 - 3P_\infty$. Denote $y = A(x)$ the hyperplane (parabola or straight line) passing through the three points. The roots of $F - A^2$ are the x-coordinates of the intersection points between the hyperplane and C. $F - A^2$ is a polynomial of degree 5, hence there are 5 intersection points, denote Q_1, Q_2 the remaining two, and define $S = Q_1 + Q_2 - 2P_\infty$, then the result is $-S$.

If $\text{weight}(R) = 4$, then $R = P_1 + P_2 + P_3 + P_4 - 4P_\infty$. Denote $y = A(x)$ the hyperplane interpolating the four points, this is a polynomial of degree at most 3. $F - A^2$ is a polynomial of degree 5 or 6, and we know 4 of the roots. Construct S as in the previous case, and the result is $-S$.

Harley implemented the algorithm using relatively fast techniques such as CRT, Newton's iteration, and Karatsuba method for polynomials multiplication. In this way the polynomial arithmetic was reduced to arithmetic over the base field.

2.3.3. *Improvements for genus two.* Since Harley's algorithm many improvements were suggested for genus 2. The main ideas of generalizations and improvements were in the following issues:

Generalization to arbitrary characteristic [Lan01, Lan02a, SMCT02, SMCT03].

Reduce the number of operations [MCT01, MDMCT02, Tak02].

Use of different sets of coordinates ([MDMCT02, Lan02b, Lan02c]).

Focus on special families of curves [PWP03, PWP04a, BD04].

The comparison between various algorithms for hyperelliptic curves of genus 2 can be found in Appendix A.

2.3.4. *Improvements for genus three.* Genus 3 hyperelliptic curves were considered to be attractive because they are resilient against Gaudry's attack on one hand, and allow working over smaller fields than genus 2 curves do. However, a recent variant of Gaudry's attack ([GTTD07]) appears to be faster than Pollard Rho attack in this case, so one should be careful about using these curves in standard cryptosystems. Genus 3 hyperelliptic curves are C_{27} curves, i.e. of the form

$$y^2 + (h_3x^3 + h_2x^2 + h_1x + h_0)y = x^7 + f_6x^6 + f_5x^5 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0.$$

In [KGMCT02] Kuroki et al. generalized Harley's algorithm to genus 3 over odd characteristic. In [PWGP03] Pelzl et al. generalized it to arbitrary characteristic. In [GMACT04] the authors managed to save some multiplications by using Toom's polynomial multiplication instead of Karatsuba's multiplication. In [FW04] Fan and Wang implemented the algorithm over odd characteristic using projective coordinates. In [FWW05] Fan et al. focused on the doubling operation over binary fields. In [ATW06] Avanzi et al. focused on the arithmetic over binary fields.

The comparison between various algorithms for hyperelliptic curves of genus 3 can be found in Appendix B.

2.3.5. *Improved Algorithms for Genus four.* Genus 4 hyperelliptic curves allow working over smaller fields than genus 2 and 3 curves. However, these curves

are less attractive for standard cryptosystems due to recent attacks ([GTTD07]). Genus 4 hyperelliptic curves are C_{29} curves, i.e. of the form

$$y^2 + (h_4x^4 + h_3x^3 + h_2x^2 + h_1x + h_0)y = x^9 + f_8x^8 + f_7x^7 + \ldots + f_1x + f_0.$$

In [PWP04b] Pelzl et al. introduced the first explicit formulae for genus 4 hyperelliptic curves. In [ATW06] Avanzi et al. used the same methods both for genus 3 and for genus 4. The authors focused on curves over binary fields, and implemented the arithmetic "tricks" that were used for smaller genus as well as more efficient field arithmetic.

The comparison between various algorithms for hyperelliptic curves of genus 4 can be found in Appendix C.

2.4. **Superelliptic Curves.** A *superelliptic curve* C is a curve that admits an affine model of the form

$$y^a = F(x) = f_bx^b + f_{b-1}x^{b-1} + \ldots + f_1x + f_0.$$

We can see that superelliptic curves are C_{ab} curves where $c_{ij} = 0$ for $0 \leq i \leq b$ and $0 < j \leq a-1$, $c_{ia} = 0$ for $0 < i \leq b$, and $c_{0a} = 1$.

In order for the superelliptic curve to be nonsingular in the affine plane we assume that $\gcd\big(F(x), F'(x)\big) = 1$, and that $\mathrm{char}(\Bbbk) \nmid a$. To ensure one and only one point at infinity we assume that $\gcd(a, b) = 1$. The field extension $\Bbbk(C)/\Bbbk(x)$ is a Galois extension, and the Galois group is $\mathrm{Gal}\big(\Bbbk(C)/\Bbbk(x)\big) = \langle\sigma\rangle$, where σ is of the form $\sigma(x, y) \mapsto (x, \zeta y)$ and ζ is a primitive a-th root of unity.

2.4.1. *GPS Algorithm.* In [GPS02] Galbraith et al. proposed an algorithm for computing in the Jacobian of superelliptic curves by adopting an LLL-like algorithm for lattice reduction to provide the reduction method. Their approach is analogous to the strategy of computing with ideals in number fields ([Coh93] Section 6.5). This algorithm belongs to the third type, Gröbner basis manipulation.

In [Pau98] Paulus modified the LLL algorithm to compute a reduced basis of a lattice in a function field. First we embed $\Bbbk[C]$ into $\Bbbk[x]^a$ in the following way:

$$\varphi : \sum_{i=1}^{a} h_i(x)y^i \mapsto \big(h_1(x), \ldots, h_a(x)\big).$$

Denote $A = \big(h_1(x), \ldots, h_a(x)\big) \in \Bbbk[x]^a$, we can define a metric on A by setting $|A|_i := \deg_x(h_i(x)) + \frac{b}{a}i$, and then $|A| := \max_i\{|A|_i\}$.

Consider an ideal $\mathfrak{a} \subseteq \Bbbk[C]$, and let $[\alpha_1, \ldots, \alpha_a]$ be a $\Bbbk[x]$-basis of $\mathfrak{a}$, then the image of $\mathfrak{a}$ under φ is a lattice over $\Bbbk[x]$ generated by $\{\varphi(\alpha_i)\}$.

The authors represented divisors classes using the unique HNF (Hermite normal form) of the reduced ideal. Given reduced ideals $\mathfrak{a}_1, \mathfrak{a}_2$, the algorithm consists of four steps

(1) $\mathfrak{b} \leftarrow \mathfrak{a}_1\mathfrak{a}_2$.
(2) $\mathfrak{c} \leftarrow$ the semi-reduced ideal equivalent to $\mathfrak{b}^{-1}$.
(3) $b \leftarrow$ a minimal nonzero element in $\mathfrak{c}$.
(4) $\mathfrak{a}_3 \leftarrow$ the HNF of $b\mathfrak{c}^{-1}$.

The complexity of the GPS algorithm is $O\big(a^6b^2g^2\big)$ operations in $\Bbbk$.

2.4.2. *Cantor's Algorithm for Superelliptic Cubics.* In [BEFG04a] Basiri et al. focused on superelliptic cubics, i.e. superelliptic curves with $a = 3$. In this case we have that $\mathrm{Gal}\big(\Bbbk(C)/\Bbbk(x)\big) = \{\mathrm{Id}, \sigma, \sigma^2\}$, where σ is of the form $\sigma(x, y) \mapsto (x, \zeta y)$ and ζ is a primitive third root of unity.

Let D be a $\Bbbk$-rational divisor. The *conjugates* of D are D^σ and D^{σ^2}. The authors noticed that Mumford's representation is suitable for a special class of divisors, namely divisors that do not have any two conjugate points in their support. A divisor D is called *typical* if it is of the form $D = \mathrm{div}(U, y - V)$ for some $U, V \in \Bbbk[x]$ such that $\deg(V) < \deg(U) \leq g$ and $U \mid (V^3 - F)$. For a fixed g, the probability that a reduced divisor is not typical is $O(\frac{1}{q})$.

Given a superelliptic cubic of genus 3 or 4, a typical divisor $\mathrm{div}(U, y - V)$ is reduced whenever $\deg(U) < g$, or $\deg(U) = g$ and $\deg(V) = g - 1$.

Based on the similarity between Mumford's representation of reduced divisors on a hyperelliptic curve and the representation of typical divisors on a superelliptic cubic, Basiri et al. generalized Cantor's algorithm to the case of superelliptic cubics. The classic approach of Gauß' reduction fails for superelliptic cubics, so Lagrange reduction is used in this case.

2.4.3. *Bauer's Algorithm for Superelliptic Cubics.* In [Bau03] Bauer implemented GPS algorithm for superelliptic curves in the specific case of superelliptic cubics. The author noticed that because GPS algorithm is very general it contains certain inefficiencies, and by exploiting the underlying structure of superelliptic cubics the arithmetic can be improved.

2.4.4. *Flon-Oyono Algorithm for Picard Curves.* A *Picard curve* is a superelliptic curve of genus 3: $y^3 = F(x) = x^4 + f_3x^3 + f_2x^2 + f_1x + f_0$.

In [FO04] Flon and Oyono took the approach of hyperplane intersection, and obtained explicit formulae for computing in the Jacobian of Picard curves. Following [BEFG04a], the authors concentrated on addition in the most frequent cases, i.e. of typical divisors.

Given two reduced divisors $D_1 = P_1 + P_2 + P_3 - 3P_\infty$ and $D_2 = Q_1 + Q_2 + Q_3 - 3P_\infty$, we want to find the reduced divisor equivalent to $P_1 + P_2 + P_3 + Q_1 + Q_2 + Q_3 - 6P_\infty$. For this we look at the divisor

$$D = -\big(P_1 + P_2 + P_3 + Q_1 + Q_2 + Q_3 - 9P_\infty\big).$$

D is a $\Bbbk$-rational divisor of degree 3, so by Riemann-Roch there exists a function $W \in \Bbbk(C)^*$ such that $\mathrm{div}(W) \geq -D$. The only pole of W is P_∞, hence $W \in \Bbbk[C]$. In addition, $\mathrm{ord}_{P_\infty}(W) \geq -9$, so W is an element of the $\Bbbk$-vector space spanned by $1, x, x^2, xy, y^2, x^3$. Take W to be the unique such element with max. order at P_∞.

If W is a conic, then $\mathrm{Supp}(D_1 + D_2)$ consists of 6 points aside from P_∞ that lie on W. This conic intersects C in exactly two more points R_1 and R_2. Taking a hyperplane through these points gives us two new points K_1, K_2. Thus, the reduction of $D_1 + D_2$ is $K_1 + K_2 - 2P_\infty$.

If W is a cubic, then by Bézout theorem W intersects C in exactly three more points R_1, R_2, R_3. We get that

$$(P_1 + P_2 + P_3 - 3P_\infty) + (Q_1 + Q_2 + Q_3 - 3P_\infty) = -(R_1 + R_2 + R_3 - 3P_\infty) + \mathrm{div}(W).$$

Using Riemann-Roch again we get that there exists a unique conic W_2 passing through R_1, R_2, R_3 and through three more points K_1, K_2, K_3. Thus, $(K_1 + K_2 + K_3 - 3P_\infty)$ is the reduced representative of $D_1 + D_2$.

2.5. C_{ab} **Curves.** In this section we review some generic algorithms for C_{ab} curves, and focus on the more cryptographically interesting C_{34} curves.

2.5.1. *Arita's Algorithm for C_{ab} Curves.* In [Ari99] Arita proposed an addition algorithm in the Jacobian of C_{ab} curves. Reduced ideals are represented by their reduced Gröbner basis w.r.t. C_{ab} order.

Let C be a C_{ab} curve, and let $\mathfrak{a} \subset \Bbbk[C]$ be an ideal. We denote by $f_{\mathfrak{a}}$ the nonzero 'monic' polynomial with smallest leading monomial (w.r.t. C_{ab} order) in $\mathfrak{a}$, and $\mathfrak{a}^* := \big(\langle f_{\mathfrak{a}}\rangle :_{\Bbbk[C]} \mathfrak{a}\big) = \big\{g \in \Bbbk[C] : g\mathfrak{a} \subseteq \langle f_{\mathfrak{a}}\rangle\big\}$. Notice that $\mathfrak{a}\mathfrak{a}^* = \langle f_{\mathfrak{a}}\rangle$, thus $\mathfrak{a}^* = \mathfrak{a}^{-1}$. An ideal $\mathfrak{a}$ is reduced iff $\mathfrak{a} = \mathfrak{a}^{**}$.

Given two reduced ideals $\mathfrak{a}_1, \mathfrak{a}_2 \subseteq \Bbbk[C]$, in the composition step we compute the ideal $\mathfrak{b} = \mathfrak{a}_1\mathfrak{a}_2$. In the reduction step we first compute the reduced inverse $\mathfrak{b}^*$ and later the reduced ideal $\mathfrak{a}_3 = \mathfrak{b}^{**}$. We need to find the polynomial $g = f_{\mathfrak{b}}$ such that $\langle g\rangle = \mathfrak{b}\mathfrak{b}^*$ and the polynomial $h = f_{\mathfrak{b}^*}$ such that $\langle h\rangle = \mathfrak{b}^*\mathfrak{b}^{**}$. Combining these relations we get that $\mathfrak{b}\langle h\rangle = \mathfrak{b}\mathfrak{b}^*\mathfrak{b}^{**} = \langle g\rangle\mathfrak{b}^{**}$, hence $\mathfrak{b}^{**} = \frac{h}{g}\mathfrak{b}$.

(1) $\mathfrak{b} \leftarrow \mathfrak{a}_1\mathfrak{a}_2$.
(2) $g \leftarrow$ the minimal nonzero element in $\mathfrak{b}$ w.r.t. C_{ab} order.
(3) $h \leftarrow$ the minimal nonzero element w.r.t. C_{ab} order, satisfying $h\mathfrak{b} \subseteq \langle g\rangle$.
(4) $\mathfrak{a}_3 \leftarrow \frac{h}{g}\mathfrak{b}$.

The minimal elements in steps 2 and 3 are found by computing the reduced Gröbner bases of the ideals $\mathfrak{b}$ and $\mathfrak{b}^*$. Arita used Buchberger's method for this computation. The complexity of this algorithm is $O(g^3 \log^2 q)$.

2.5.2. *GPS Algorithm for C_{ab} Curves.* In [HS00] Harasawa and Suzuki generalized the GPS algorithm of superelliptic curves to C_{ab} curves. In order to generalize GPS algorithm, the authors had to address two issues:

(1) Given an ideal $\mathfrak{a}$, how to compute the inverse $\mathfrak{a}^{-1}$.
(2) How to compute the minimal element over an ideal w.r.t. C_{ab} order.

The first issue is treated with methods of computing inverse ideals in the integral closure of a number field (see [Coh93, Prop. 4.8.19]). In the case of C_{ab} curves, $\Bbbk[C]$ is the integral closure of $\Bbbk[x]$ in $\Bbbk(C)$, and the set $\{1, y, \ldots, y^{a-1}\}$ is a $\Bbbk[x]$-basis of $\Bbbk[C]$.

Regarding the second issue, given $h = \sum_{i=1}^{a} h_i(x)y^i \in \Bbbk[C]$, by the definition of the metric $|\cdot|$, we have

$$-\mathrm{ord}_{P_\infty}(h) = \max_{1\le i\le a}\Big\{a\deg_x(h_i) + bi\Big\} = a\max_{1\le i\le a}\Big\{\deg_x(h_i) + \frac{b}{a}i\Big\} = a|\varphi(h)|.$$

Therefore, for an ideal $\mathfrak{a} \subseteq \Bbbk[C]$, finding the minimal element over $\mathfrak{a}$ w.r.t. the C_{ab} order is the same as finding the minimal element over $\varphi(\mathfrak{a})$ w.r.t. the metric $|\cdot|$. Thus we can apply Paulus' method for lattice reduction in order to find the minimal element w.r.t. the C_{ab} order.

The overall complexity Harasawa and Suzuki obtained for the addition in the Jacobian of a C_{ab} curve is $O(a^8 g^2 \log^2 q)$.

2.5.3. *Arita's Algorithm for C_{34} Curves.* In [Ari01] Arita managed to simplify his algorithm in the case of C_{34} curves. This was accomplished by classifying the Gröbner bases of the ideals, and so computing their Gröbner bases without the use of Buchberger algorithm. The author carried out the computation symbolically and managed to obtained explicit formulae.

The genus of a C_{34} curve is 3, therefore, after the composition step the order of the ideals is at most 6. The minimal six polynomials w.r.t. C_{34}-order are $M = \{1, x, y, x^2, xy, y^2\}$. Arita classified the ideals of degree at most 6 based on the different possibilities of linear independence of polynomials in M.

2.5.4. *BEFG Algorithm for C_{34} Curves.* In [BEFG04a] and [BEFG04b] the authors managed to obtain explicit formulae for adding and doubling typical reduced ideals in C_{34} curves. The underlying method for the reduction step was the FGLM algorithm for switching between Gröbner bases of different orderings, and so compute the Gröbner basis in a C_{34} order from the Gröbner basis in lexicographic order.

Let C be a C_{34} curve of the form $y^3 + H(x)y = F(x)$ where $\deg(F) = 4$ and $\deg(H) \leq 2$. In the composition step, given two typical ideals $\mathfrak{a}_i = \langle U_i, y - V_i \rangle$ where $\deg(U_i) = 3$ and $\deg(V_i) = 2$, we get the product $\mathfrak{b} = \mathfrak{a}_1\mathfrak{a}_2 = \langle U, y - V \rangle$ where $U = U_1U_2$ of degree 6 and $\deg(V) = 5$. In the case of addition, where $U_1 \neq U_2$ and $\gcd(U_1, U_2) = 1$, we can find V using the CRT as follows:

$$S_1 \equiv U_1^{-1} \pmod{U_2}, \quad T \equiv S_1(V_2 - V_1) \pmod{U_2}, \quad V = V_1 + TU_1.$$

In the reduction step we have the ideal $\mathfrak{b} = \langle U, y - V \rangle$. Let E be the minimal element w.r.t. C_{34} order in the ideal $\mathfrak{b}^{-1} = \langle U, y^2 + Vy + V^2 + H \rangle$. The reduced ideal is $\mathfrak{a}_3 = \frac{E}{U}\mathfrak{b} = \langle U_3, y - V_3 \rangle$.

2.5.5. *FOR Algorithm for C_{34} Curves.* In [FOR04] Flon et al. focused on Jacobians of non-hyperelliptic curves of genus 3. Such a curve can be represented as a smooth projective plane quartic C. We assume there exists a rational line ℓ^∞ which crosses C in four $\mathbb{k}$-rational points P_1^∞, P_2^∞, P_3^∞ and P_4^∞. There are 5 possibilities:

(1) The four points are distinct.
(2) $P_1^\infty = P_2^\infty$, then ℓ^∞ is tangent to C at P_i^∞.
(3) $P_1^\infty = P_2^\infty = P_3^\infty$, then the point P_1^∞ is called a *flex*.
(4) $P_1^\infty = P_2^\infty$ and $P_3^\infty = P_4^\infty$, then ℓ^∞ is call *bitangent*.
(5) $P_1^\infty = P_2^\infty = P_3^\infty = P_4^\infty$, then the point P_1^∞ is called a *hyperflex*.

Denote $D^\infty = P_1^\infty + P_2^\infty + P_3^\infty$. For every $D \in \mathrm{Div}_{\mathbb{k}}^0(C)$ let D^+ be an effective divisor such that $D^+ - D^\infty \sim D$. The algorithm is based on the following theorem.

Theorem 2.5.1 ([FOR04] p. 4)**.** *Let $D_1, D_2 \in Div_{\mathbb{k}}^0(C)$. Then $D_1 + D_2$ is equivalent to a divisor $D = D^+ - D^\infty$, where the points in $Supp(D^+)$ are given by the following:*

(1) *Take the unique cubic E which goes (with multiplicity) through the support of D_1^+, D_2^+ and P_1^∞, P_2^∞ and P_4^∞. This cubic also crosses C in the residual effective divisor D_3.*
(2) *Take the unique conic Q which goes through the support of D_3 and P_1^∞, P_2^∞. This conic also crosses C in the residual effective divisor D^+.*

The authors implemented this algorithm for the flex case, with cost of 2I + 163M for addition and 2I + 185M for doubling. In the subcase of hyperflex we get a C_{34} curve, and the cost is 2I + 145M for addition and 2I + 167M for doubling.

The comparison of the algorithms for C_{34} curves can be found in Appendix D. [1]

[1] After the release of this paper another algorithm was published in [ASM07], using an approach which reduces the computation to linear algebra.

3. C_A Curves

In the previous section we described the group law operation in the Jacobian group of certain nonsingular curves. In this section we discuss a generalization of this situation, and explain the group law in the generalized Jacobian of (possibly singular) C_A curves. C_{ab} curves are a special case of C_A curves. The concept of generalized Jacobian was first introduced by Rosenlicht in [Ros54] More details can be found in [Ser88].

3.1. **Generalized Jacobian Varieties.** Let $C/\Bbbk$ be a complete irreducible nonsingular projective curve, let $\mathfrak{m} = \sum m_P P$ be an effective $\Bbbk$-rational divisor, and let $S = \mathrm{Supp}(\mathfrak{m})$. We call $\mathfrak{m}$ a *modulus*. Given a function $f \in \Bbbk(C)^*$ we denote $f \equiv 1 \pmod{\mathfrak{m}}$ if for every $P \in C$ we have $\mathrm{ord}_P(1-f) \geq m_P$.

Definition 3.1.1 ($\mathfrak{m}$-equivalence)**.** Let D_1 and D_2 be two divisors over C prime to S. We say that D_1 and D_2 are $\mathfrak{m}$-*equivalent*, and write $D_1 \sim_{\mathfrak{m}} D_2$, if there is a function $f \in \Bbbk(C)^*$ such that $\mathrm{div}(f) = D_1 - D_2$ and $f \equiv 1 \pmod{\mathfrak{m}}$.

Given an irreducible nonsingular curve C, a finite subset of $S \subseteq C$ and an equivalence relation $\sim$ on S, one can construct a singular curve $C' = (C \backslash S) \cup (S/\sim)$ with C its normalization. Given a modulus $\mathfrak{m}$ with $\deg(\mathfrak{m}) \geq 2$, $\sim_{\mathfrak{m}}$ is an equivalence relation and we denote the singular curve C' by $C_{\mathfrak{m}}$.

Just like linear equivalence gives rise to the Jacobian variety, given a modulus $\mathfrak{m}$ the equivalence relation $\sim_{\mathfrak{m}}$ gives rise to a generalized Jacobian variety, $J_{\mathfrak{m}}$.

The dimension of $J_{\mathfrak{m}}$ is the arithmetic genus of the curve $C_{\mathfrak{m}}$, that is

$$\pi = \begin{cases} g, & \mathfrak{m} = 0 \\ g + \deg(\mathfrak{m}) - 1, & \mathfrak{m} \neq 0 \end{cases}$$

Let $\mathfrak{m}$ be a modulus on C and $S = \mathrm{Supp}(\mathfrak{m})$, we define $\mathrm{Div}_{\mathfrak{m}}(C)$ to be the subgroup of $\mathrm{Div}(C)$ formed by divisors prime to S and $\mathrm{Div}^0_{\mathfrak{m}}(C)$ its subgroup of degree zero. We denote by $\mathrm{Pic}_{\mathfrak{m}}(C)$ (and $\mathrm{Pic}^0_{\mathfrak{m}}(C)$) the quotient group of $\mathrm{Div}_{\mathfrak{m}}(C)$ (resp. $\mathrm{Div}^0_{\mathfrak{m}}(C)$) of $\mathfrak{m}$-equivalence classes. $J_{\mathfrak{m}}$ is isomorphic (as a group) to $\mathrm{Pic}^0_{\mathfrak{m}}(C)$.

In order to understand the structure of $J_{\mathfrak{m}}$ first note that there are isomorphisms of groups $\varphi : \mathrm{Pic}^0(C) \to J$ and $\psi : \mathrm{Pic}^0_{\mathfrak{m}}(C) \to J_{\mathfrak{m}}$. In addition, every pair of $\mathfrak{m}$-equivalent divisors is obviously also linearly equivalent, so we have an epimorphism $\sigma : \mathrm{Pic}^0_{\mathfrak{m}}(C) \to \mathrm{Pic}^0(C)$. Combining the three maps together, we get an epimorphism $\tau : J_{\mathfrak{m}} \to J$ defined by $\tau = \varphi \circ \sigma \circ \psi^{-1}$. Denote by $L_{\mathfrak{m}}$ the kernel of τ, and we get the following short exact sequence of groups

$$0 \longrightarrow L_{\mathfrak{m}} \hookrightarrow J_{\mathfrak{m}} \xrightarrow{\tau} J \longrightarrow 0.$$

Thus, the generalized Jacobian $J_{\mathfrak{m}}$ is an extension of the Jacobian J by $L_{\mathfrak{m}}$. The algebraic group $L_{\mathfrak{m}}$ is biregular isomorphic to $R_{\mathfrak{m}}/\mathbb{G}_m$, i.e.

$$L_{\mathfrak{m}} \simeq \prod_{i=1}^{\#S-1} \mathbb{G}_m \times \prod_{P \in S} V_{(m_P)},$$

where $V_{(m_P)}$ is a unipotent group isomorphic to a group of matrices.

3.2. **The Ideal Class Group of Singular Curves.** Let $C_0/\Bbbk$ be a plane irreducible projective curve. We fix a $\Bbbk$-rational point $P_\infty \in C_0$. Let C_1 be the curve obtained by desingularizing only the point P_∞. We assume there is only one point lying above P_∞ in C_1 which is also denoted P_∞. Let S be the set of singular

points of C_1, and let C be the normalization of C_1 (and of C_0) with the map $d : C \to C_1 \to C_0$. Denote by $\mathfrak{m}$ the modulus defined by $d^{-1}(S)$. Let g be the genus of the curve C.

We assume $C_1^a = C_1 \backslash \{P_\infty\}$ is an affine curve, and let $R = \Bbbk[C_1^a]$ be its coordinate ring. Note that if C_1^a has singular points R is not a Dedekind domain. However, we can still denote by $\mathrm{Id}(R)$ the group of invertible fractional ideals of R, and its subgroup $\mathrm{PId}(R) = \{\langle f \rangle = fR : f \in \Bbbk(C)^*\}$. Thus, we can define with this notation the ideal class group $\mathrm{IdCl}(R) = \mathrm{Id}(R)/\mathrm{PId}(R)$.

The local ring of a nonsingular point P is regular, i.e. the corresponding maximal ideal $\mathfrak{m}_P$ is principal. However, the maximal ideal corresponding to a singular point is generated by more than one element. Therefore, the group of invertible fractional ideals of R is the free abelian group generated by maximal ideals corresponding to nonsingular points. We generalize the situation of the previous section by assigning divisors prime to S, i.e. divisors generated by nonsingular points, to invertible fractional ideals generated by the corresponding regular maximal ideals, and vice-versa. For conclusion, we get the following sequence:

$$J_{\mathfrak{m}}(C) \simeq J_{\mathfrak{m}}(C_1) \simeq \mathrm{Pic}^0_{\mathfrak{m}}(C_1) \simeq \mathrm{Pic}_{\mathfrak{m}}(C_1^a) \simeq \mathrm{IdCl}(R).$$

3.3. C_A **Curves.** We denote by $\mathbb{N}$ the set of non-negative integers. Let $M \subset \mathbb{N}$ be a finitely generated semigroup, i.e.

$$M = \langle a_1, \ldots, a_t \rangle = \mathbb{N}a_1 + \ldots + \mathbb{N}a_t$$

where $t \leq a_1$, and $a_i \neq 0$. The complement of M in $\mathbb{N}$ is finite iff $\gcd(a_1, \ldots, a_t) = 1$. If we denote $b_i = \min\{a \in M : a \equiv i \pmod{a_1}\}$, then

$$\#(\mathbb{N}\backslash M) = \sum_{i=1}^{a_1 - 1} \left[\frac{b_i}{a_1} \right].$$

In this case M is called a *numerical semigroup*.

Let M be a numerical semigroup with a system of generators $A = \{a_1, \ldots, a_t\}$, where $t \leq a_1$. We now define a surjective map $\Psi : \mathbb{N}^t \longrightarrow M$ by

$$\Psi(n_1, n_2, \ldots, n_t) = \sum_{i=1}^{t} n_i a_i.$$

Using this map, we can define a monomial order on $\mathbb{N}^t$ as follows.

Definition 3.3.1 (C_A order)**.** Given $\alpha = (\alpha_1, \ldots, \alpha_t), \beta = (\beta_1, \ldots, \beta_t) \in \mathbb{N}^t$ we say that $\alpha <_A \beta$, if one of the following holds:

(1) $\Psi(\alpha) < \Psi(\beta)$;
(2) $\Psi(\alpha) = \Psi(\beta)$, and $\alpha_1 = \beta_1, \ldots, \alpha_i = \beta_i, \alpha_{i+1} > \beta_{i+1}$.

Definition 3.3.2. For $a \in M$ we define $\mathbf{m}(a) = \min\{n \in \mathbb{N}^t : n \in \Psi^{-1}(a)\}$, and as before $b_i = \min\{a \in M : a \equiv i \pmod{a_1}\}$. In addition we set
$B(A) = \{\mathbf{m}(a) : a \in M\} \subset \mathbb{N}^t$.
$T(A) = \{\mathbf{m}(b_i) \in B(A) : i = 0, \ldots, a_1 - 1\}$.
$V(A) = \{\ell \in \mathbb{N}^t \backslash B(A) : \text{if } \ell = m + n \text{ with } m \in \mathbb{N}^t \backslash B(A) \text{ and } n \in \mathbb{N}^t, \text{ then } n = 0\}$.

$V(A)$ is a finite set and $B(A) = T(A) + \mathbb{N} \times \{0\}^{t-1}$.

Under these notations we consider polynomials $F_m \in \Bbbk[x_1, \ldots, x_t]$, for $m \in V(A)$, satisfying the following conditions:

(D1) For each $m \in V(A)$ set $\ell = \mathbf{m}\big(\Psi(m)\big)$, then

$$F_m = X^m + a_\ell X^\ell + \sum_n a_n X^n,$$

where $a_\ell \neq 0$, and the sum runs over $n \in B(A)$ with $n < \ell$.

(D2) $\big\{\sum_{n\in B(A)} \Bbbk X^n\big\} \bigcap \big\{F_m : m \in V(A)\big\} = \big\{0\big\}$.

The notation X^m means $x_1^{m_1} \cdots x_t^{m_t}$, where $m = (m_1, \ldots, m_t)$.

Let $C/\Bbbk$ be an irreducible nonsingular curve. Let $P \in C$ be a $\Bbbk$-rational point, like in C_{ab} curves we define

$$M(P) = \big\{ -\operatorname{ord}_P(f) : f \in \mathcal{L}(\infty P)\backslash\{0\}\big\} \subset \mathbb{N},$$

then $M(P)$ is a numerical semigroup.

Let R be a subalgebra of $\mathcal{L}(\infty P)$ such that $\Bbbk \subset R \subset \mathcal{L}(\infty P)$, and define a semigroup by

$$M(R) = \big\{ -\operatorname{ord}_P(f) : f \in R\backslash\{0\}\big\} \subset \mathbb{N}.$$

Lemma 3.3.3 ([AMS04], Lemma 3.1)**.** *The field of fractions of R coincides with $\Bbbk(C)$ iff $M(R)$ is a numerical semigroup.*

Hereafter we assume that $M(R)$ is a numerical semigroup. For every $i = 1, \ldots, t$ we choose a function $f_i \in R$ such that $\operatorname{ord}_P(f_i) = -a_i$, and consider the surjection $\Theta : \Bbbk[x_1, \ldots, x_t] \longrightarrow R$ defined by $\Theta(F) = F(f_1, \ldots, f_t)$ for $F \in \Bbbk[x_1, \ldots, x_t]$. The kernel of this map, denoted $I(R) = \ker\Theta$, is generated by polynomials satisfying conditions (D1) and (D2). Miura showed the converse is also true.

Theorem 3.3.4 ([AMS04], Theorem 3.2)**.** *Let M be a numerical semigroup with a system of generators $A = \{a_1, \ldots, a_t\}$. Then an ideal $I \subset \Bbbk[x_1, \ldots, x_t]$ is generated by polynomials satisfying conditions (D1) and (D2) iff there exist a function field $\mathbb{K}$ of one variable over $\Bbbk$, a $\Bbbk$-rational point P of the nonsingular model of $\mathbb{K}$, and a subalgebra $\Bbbk \subset R \subset \mathcal{L}(\infty P)$ such that the field of fractions of R is $\mathbb{K}$, and $I = I(R)$.*

In this case the polynomials satisfying (D1) and (D2) form a Gröbner basis of I, and the projective model $C_0(A)$ of Spec$(\Bbbk[x_1, \ldots, x_t]/I) \subset \mathbb{A}^t_\Bbbk$ in $\mathbb{P}^t_\Bbbk$ has only one infinite point P, which is at most a cuspidal singularity. Moreover, the affine model Spec$(\Bbbk[x_1, \ldots, x_t]/I)$ of $\mathbb{K}$ is nonsingular iff $R = \mathcal{L}(\infty P)$.

Definition 3.3.5 (C_A curve, [AMS04])**.** Under the notation of the theorem, when the ideal $I = I(R) \subset \Bbbk[x_1, \ldots, x_t]$ corresponds to the numerical semigroup M with a system of generators $A = \{a_1, \ldots, a_t\}$, we call Spec$(\Bbbk[x_1, \ldots, x_t]/I)$, or $C_0(A)$, a *C_A curve*. For a numerical semigroup M generated by A, if there exists an affine nonsingular C_A curve, M is called a Weierstraß numerical semigroup.

Using the Riemann-Roch theorem for singular curves we obtain the following formula for the genus of a C_A curve.

Proposition 3.3.6 ([AMS04], Prop. 3.3)**.** *Let C_1 be the curve given by desingularizing only the point at infinity of a C_A curve $C_0(A)$. The arithmetic genus of C_1 is*

$$p_a(C_1) = \#(\mathbb{N}\backslash M) = \sum_{i=1}^{a_1-1} \left[\frac{b_i}{a_1}\right].$$

If a C_A curve is generated by two coprime elements, $A = \{a, b\}$, we obtain a C_{ab} curve. A beautiful example of a C_{357} curve can be found in [AMS04].

3.4. **Arita's Algorithm for C_A Curves.** In [AMS04] the authors generalized Arita's algorithm for C_{ab} curves to the case of C_A curves. Let C be a C_A curve, and following the notations of Theorem 3.3.4 let $R = \Bbbk[C] = \Bbbk[x_1, \ldots, x_t]/I$ be its coordinate ring, and consider the canonical map $\Theta : \Bbbk[x_1, \ldots, x_t] \to R$ whose kernel is I. Note that C_A order is a monomial order on $\Bbbk[x_1, \ldots, x_t]$. For every ideal $\mathfrak{a} \subset R$ we denote $\mathfrak{A} = \Theta^{-1}(\mathfrak{a})$.

Let $\mathfrak{a} \subset R$ be an invertible ideal, denote $h \in \left(1 :_{\Bbbk(C)} \mathfrak{a}\right) = \mathfrak{a}^{-1}$ the nonzero element of $\mathfrak{a}^{-1}$ with smallest minus order $-\mathrm{ord}_{P_\infty}(h)$, and define $\mathfrak{a}^* = h\mathfrak{a}$.

Lemma 3.4.1 ([AMS04])**.** *For an invertible ideal $\mathfrak{a}^*$ of R, a nonzero element h of $\mathfrak{a}$ with smallest minus order $-ord_{P_\infty}(h)$ is unique up to constant multiplication.*

It follows that given an ideal class $[\mathfrak{a}]$ the ideal $\mathfrak{a}^*$ is unique, this is the reduced representative of the ideal class.

In order to find the element $h \in \mathfrak{a}^{-1}$ we take an element $f \in \mathfrak{a}$ and find $g \in \left(\langle f\rangle :_{\Bbbk(C)} \mathfrak{a}\right) = \left(\langle f\rangle :_R \mathfrak{a}\right) = f\mathfrak{a}^{-1}$ with smallest minus order $-\mathrm{ord}_{P_\infty}(g)$, then $h = \frac{g}{f}$. The element g is found by computing the Gröbner basis w.r.t. C_A order of the ideal $\Theta^{-1}(f\mathfrak{a}^{-1}) = \left((f\Bbbk[x_1, \ldots, x_t] + I) :_{\Bbbk[x_1,\ldots,x_t]} \mathfrak{A}\right)$.

Given two reduced invertible ideals $\mathfrak{a}_1, \mathfrak{a}_2 \subseteq R$, represented with the Gröbner bases $\mathfrak{a}_1 = \langle f_1, \ldots, f_l\rangle, \mathfrak{a}_2 = \langle g_1, \ldots, g_m\rangle$:

(1) $\mathfrak{A}_1 \leftarrow \langle f_1, \ldots, f_l\rangle + I, \quad \mathfrak{A}_2 \leftarrow \langle g_1, \ldots, g_m\rangle + I.$
(2) $\mathfrak{B} \leftarrow \mathfrak{A}_1\mathfrak{A}_2.$
(3) Take a nonzero element $f \in \mathfrak{B}\backslash I$.
(4) Take $g \in \left((f\Bbbk[x_1, \ldots, x_t] + I) :_{\Bbbk[x_1,\ldots,x_t]} \mathfrak{B}\right)$ with smallest C_A order.
(5) Compute Gröbner basis of $\mathfrak{B}^* = \frac{g}{f}\mathfrak{B}$.

4. Conclusions

The Arita-Miura-Sekiguchi algorithm for C_A curves does not attempt to replace the faster algorithms for C_{ab} curves, but provides a way to work with more general curves than C_{ab} curves. Nevertheless, it is interesting to see the difference in performance between the AMS algorithm and the fast C_{ab} curves algorithms. AMS algorithm relies on Buchberger's algorithm to find the reduced Gröbner Basis, and this algorithm is hard for analysis, therefore, it is difficult to perform a theoretical comparison. In this work we implemented some of the C_{ab} curves algorithms and compared them to AMS C_A curves algorithm. The purpose of this implementation is to provide a feeling of the running time.

All computations were performed using Intel Pentium 4 CPU 3.00GHz, 1GB RAM, using Magma V2.11-14.

Examples comparing C_{34} curves can be found in Appendix E. The results show that operations that take 0.016 seconds using the FOR and BEFG algorithms take 0.125 seconds using AMS algorithm. This means that AMS algorithms is approximately 8 times slower than the FOR and BEFG algorithms for C_{34} curves.

Acknowledgment

The results of this paper were obtained during my Master studies at Bar-Ilan University. I would like to express deep gratitude to my supervisor Professor Boris Kunyavskii whose guidance and support were crucial for the successful completion of this project.

References

[ASM07] F. Abu Salem, K. Makdisi, *Fast Jacobian group operations for C_{34} curves over a large finite field*, LMS J. Comput. Math., **Vol. 10** (2007), 307-328.

[Ari99] S. Arita, *Algorithms for computations in Jacobian group of C_{ab} curve and their application to discrete-log-based public key cryptosystems*, The Institute of Electronics, Information and Communication Engineers - IEICE Trans. Fundamentals, **Vol. J82-A, No. 8** (1999), 1291-1299.

[Ari01] S. Arita, *An addition algorithm in Jacobian of C_{34} curve*, In Information Security and Privacy, ACISP 2003, Springer-Verlag, LNCS 2727, 2003, pp. 93-105.

[AMS04] S. Arita, S. Miura, T. Sekiguchi, *An addition algorithm on the Jacobian varieties of curves*, J. Ramanujan Math. Soc., **Vol. 19, No. 4** (2004), 1-17.

[ATW06] R. Avanzi, N. Theriault, Y. Wang, *Rethinking low genus hyperelliptic Jacobian arithmetic over binary fields: interplay of field arithmetic and explicit formulae*, http://www.cacr.math.uwaterloo.ca/techreports/2006/cacr2006-07.pdf, 2006.

[BEFG04a] A. Basiri, A. Enge, J.C. Faugère, N. Gürell, *The arithmetic of Jacobian groups of superelliptic cubics*, Math. of Comp., **Vol. 74, No. 249** (2004), 389-410.

[BEFG04b] A. Basiri, A. Enge, J.C. Faugère, N. Gürell, *Implementing the arithmetic of C_{34} curves*, ANTS-VI, Springer-Verlag, LNCS 3076, 2004, pp. 87-101.

[Bau03] M. Bauer, *The arithmetic of certain cubic function fields*, Math. of Comp., **Vol. 73, No. 245** (2003), 387-413.

[BD04] B. Byramjee, S. Duqesne, *Classification of genus two curves over $\mathbb{F}_{2^n}$ and optimization of their arithmetic*, Cryptology ePrint Archive, Report 2004/107, 2004, http://eprint.iacr.org.

[Can87] D. Cantor, *Computing in the Jacobian of a hyperelliptic curve*, Math. of Comp., **Vol. 48** (1987), 95-101.

[Coh93] H. Cohen, *A course in computational algebraic number theory*, Graduate texts in Math. 138, Springer-Verlag, Berlin, 1993.

[CMO98] H. Cohen, A. Miyaji, T. Ono, *Efficient elliptic curve exponentiation using mixed coordinates*, In Proceedings of ASIACRYPT 1998, LNCS 1514, 1998, pp. 51-65.

[FW04] X. Fan, Y. Wang, *Inversion free arithmetic on genus 3 hyperelliptic curves*, Cryptology ePrint Archive, Report 2004/223, 2004, http://eprint.iacr.org.

[FWW05] X. Fan, T. Wollinger, Y. Wang, *Efficient doubling on genus 3 curves over binary fields*, Topics in Cryptology - CT-RSA 2006, Springer-Verlag, LNCS 3860 (2006), pp. 64-81.

[FO04] S. Flon, R. Oyono, *Fast arithmetic on Jacobians of Picard curves*, Public Key Cryptography - PKC 2004, Springer-Verlag, LNCS 2947 (2004), pp. 55-68.

[FOR04] S. Flon, R. Oyono, C. Ritzenthaler *Fast addition on non-hyperelliptic genus 3 curves*, Cryptology ePrint Archive, Report 2004/118, 2004, http://eprint.iacr.org.

[GPS02] S. Galbraith, S. Paulus, N. Smart, *Arithmetic on superelliptic curves*, Math. of Comp., **Vol. 71, No. 237** (2002), 393-405.

[Gau00] P. Gaudry, *An algorithm for solving the discrete log problem on hyperelliptic curves*, EUROCRYPT 2000, Springer-Verlag, LNCS 1807, 2000, pp. 19-34.

[GTTD07] P. Gaudry, N. Thériault, E. Thomé, C. Diem, *A double large prime variation for small genus hyperelliptic index calculus*, , Math. of Comp., **Vol. 76** (2007), 475-492.

[GMACT04] M. Gonda, K. Matsuo, K. Aoki, J. Chao, S. Tsujii, *Improvements of addition algorithm on genus 3 hyperelliptic curves and their implementation*, In Proc. of SCIS 2004, Japan, 2004.

[HS00] R. Harasawa, J. Suzuki, *Fast Jacobian group arithmetic on C_{ab} curves*, ANTS-IV, Springer-Verlag, LNCS 1838, 2000, pp. 359-376.

[Har00a] R. Harley, *Fast addition on genus two hyperelliptic curves*, http://cristal.inria.fr/∼harley/hyper/adding.text, 2000.

[Har00b] R. Harley, *Fast doubling on genus two hyperelliptic curves*, http://cristal.inria.fr/∼harley/hyper/doubling.c, 2000.

[Kob89] N. Koblitz, *Hyperelliptic cryptosystems*, J. of Cryptology, 1 (1989), 139-150.

[KGMCT02] J. Kuroki, M. Gonda, K. Matsuo, J. Chao, S. Tsujii, *Fast genus three hyperelliptic curve cryptosystems*, In Proc. of SCIS2002, IEICE Japan, pp. 503-507, 2002.

[Lan01] T. Lange, *Efficient arithmetic on hyperelliptic curves*, PhD thesis, University Essen, 2001.

[Lan02a] T. Lange, *Efficient arithmetic on genus 2 hyperelliptic curves over finite fields via explicit formulae*, Cryptology ePrint Archive, Report 2002/121, 2002, http://eprint.iacr.org.

[Lan02b] T. Lange, *Inversion free arithmetic on genus 2 hyperelliptic curves*, Cryptology ePrint Archive, Report 2002/147, 2002, http://eprint.iacr.org.

[Lan02c] T. Lange, *Weighted coordinates on genus 2 hyperelliptic curves*, Cryptology ePrint Archive, Report 2002/153, 2002, http://eprint.iacr.org.

[MCT01] K. Matsuo, J. Chao, S. Tsujii, *Fast genus two hyperelliptic curve cryptosystems*, Technical report of IEICE, ISEC2001-31, 2001, pp. 89-96.

[MDMCT02] Y. Miamoto, H. Doi, K. Matsao, J. Chao, S. Tsujii, *A fast addition algorithm of genus two hyperelliptic curve*, The 2002 Symp. on Cryptography and Info. Security, Japan, SCIS, pp. 497-502, 2002.

[Mum84] D. Mumford, *Tata lectures on theta II - Jacobian theta functions and differential equations*, In Prog. Math. Volume 43, Birkhäuser, 1984.

[Nag00] K. Nagao, *Improving group law algorithm for Jacobians of hyperelliptic curves*, W. Bosma ed., ANTS-IV, Springer-Verlag, LNCS 1838, 2000, pp. 439-448.

[Pau98] S. Paulus, *Lattice basis reduction in function fields*, In J. Buhler ed., Proceedings of ANTS-III, Springer-Verlag, LNCS 1423 (1998), pp. 567-575.

[PWGP03] J. Pelzl, T. Wollinger, J. Guajardo, C. Paar, *Hyperelliptic curve cryptosystems: closing the performance gap to elliptic curves*, C. Walter, Ç. Koç, and C. Paar, ed., CHES, Springer-Verlag, LNCS 2779 (2003), pp. 349-365.

[PWP03] J. Pelzl, T. Wollinger, C. Paar, *High performance arithmetic for hyperelliptic curve cryptosystems of genus two*, Cryptology ePrint Archive, Report 2003/212, 2003, http://eprint.iacr.org.

[PWP04a] J. Pelzl, T. Wollinger, C. Paar, *High performance arithmetic for special hyperelliptic curve cryptosystems of genus two*, International Conference on Information Technology: Coding and Computing - ITCC 2004, **Vol. 2**, 2004, pp. 513-557.

[PWP04b] J. Pelzl, T. Wollinger, C. Paar, *Low cost security: explicit formulae for genus 4 hyperelliptic curves*, Selected Areas in Cryptography SAC 2003, Springer-Verlag, LNCS 3006 (2004), pp. 1-16.

[Ros54] M. Rosenlicht, *Generalized Jacobian varieties*, The Annals of Math., 2nd Ser., **Vol. 59, No. 3** (1954), 505-530.

[Ser88] J P. Serre, *Algebraic groups and class fields*, Springer-Verlag, 1988.

[Sil86] J.H. Silverman, *The arithmetic of elliptic curves*, Springer-Verlag, New York, 1986.

[SMCT02] H. Sugizaki, K. Matsuo, J. Chao, S. Tsujii, *An extension of Harley addition algorithm for hyperelliptic curves over finite fields of characteristic two*, IEICE Technical report, ISEC2002-9, pp. 49-56, 2002.

[SMCT03] H. Sugizaki, K. Matsuo, J. Chao, S. Tsujii, *A generalized Harley algorithm for genus two hyperelliptic curves*, In Proc. of SCIS2003, IEICE Japan, pp. 917-921, 2003.

[Tak02] M. Takahashi, *Improving Harley algorithms for Jacobians of genus 2 hyperelliptic curves*, In Proc. of SCIS2002, IEICE Japan, pp. 155-160, 2002.

Appendix A. Genus 2 hyperelliptic curves – summary

The following table compares various algorithms for genus 2 hyperelliptic curves. The hyperelliptic curve is of the form

$$y^2 + (h_2x^2 + h_1x + h_0)y = x^5 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0.$$

Algorithm	char($\Bbbk$)	Properties	Addition	Doubling
1987 Cantor	general		3I + 70M/S	3I + 76M/S
2000 Nagao	odd	regular representation	2I + 52M/S	2I + 59M/S
	odd	alternative representation	I + 56M/S	I + 66M/S
2000 Harley	odd		2I + 24M + 3S	2I + 30M/S
2001 Lange	general		2I + 24M + 3S	2I + 26M + 6S
2001 MCT	odd		2I + 22M + S	2I + 23M + 2S
2002 MDMCT	odd	affine, $f_4 = 0$	I + 24M + 2S	I + 23M + 4S
	odd	projective, $f_4 = 0$	51M + 3S	47M + 6S
2002 Takahashi	odd		I + 23M + 2S	I + 21M + 8S
2002 Lange a	general	$h_2, h_1, f_4 \in \{0, 1\}$	I + 22M + 3S	I + 22M + 5S
	even	$h_2, h_1, f_4 \in \{0, 1\}$	I + 22M + 2S	I + 20M + 5S
2002 Lange b	general	projective	47M + 4S	40M + 6S
	general	mixed	40M + 3S	
2002 Lange c	odd	weighted, $f_4 = 0$	47M + 7S	34M + 7S
	odd	mixed, $f_4 = 0$	36M + 5S	
	even	weighted, $f_4 = 0$, $h_2 \neq 0$	46M + 4S	35M + 6S
	even	mixed, $f_4 = 0$, $h_2 \neq 0$	35M + 5S	
	even	weighted, $f_4 = 0$, $h_2 = 0$	44M + 6S	29M + 6S
	even	mixed, $f_4 = 0$, $h_2 = 0$	34M + 6S	
2002 SMCT	even	$f_4 = f_2 = 0$, $h_2 = 1$	I + 23M + 2S	I + 26M + S
2003 SMCT	general		I + 28M + S	I + 38M
2003 PWP	even	$y^2 + xy = x^5 + f_1x + f_0$		I + 9M +6S
2004 BD	even	affine, general	I + 25M	I +27M
	even	affine, $\deg(H) = 2$	I + 25M	I +26M
	even	affine, $\deg(H) = 1$	I + 24M	I +18M
	even	projective, general	45M	45M
	even	projective, $\deg(H) = 2$	45M	44M
	even	projective, $\deg(H) = 1$	42M	31M
	even	modified, general	45M	43M
	even	modified, $\deg(H) = 2$	45M	42M
	even	modified, $\deg(H) = 1$	42M	31M
	even	weighted, general	42M	46M
	even	weighted, $\deg(H) = 2$	42M	45M
	even	weighted, $\deg(H) = 1$	40M	27M

Table 1. Genus 2 hyperelliptic curve arithmetic – summary

Appendix B. Genus 3 hyperelliptic curves – summary

The following table compares various algorithms for genus 3 hyperelliptic curves. The hyperelliptic curve is of the form
$y^2 + (h_3x^3 + h_2x^2 + h_1x + h_0)y = x^7 + f_6x^6 + f_5x^5 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0$.

Algorithm	char($\mathbb{k}$)	Properties	Addition	Doubling
1987 Cantor	general		4I + 200M/S	4I + 207M/S
2000 Nagao	odd	regular representation	2I + 144M/S	2I + 153M/S
	odd	alternative representation	2I + 157M/S	2I + 170M/S
2002 KGMCT	odd	$f_6 = 0$	I + 81M/S	I + 74M/S
2003 PWGP	general	$h_i \in \{0, 1\}$, $f_6 = 0$	I + 70M + 6S	I + 61M + 10S
	even	$h_i \in \{0, 1\}$, $f_6 = 0$	I + 65M + 6S	I + 53M + 10S
	even	$H(x) = 1$, $f_6 = 0$	I + 65M + 6S	I + 14M + 11S
2004 GMACT	odd	$f_6 = 0$	I + 67M + 3S	I + 61M + 8S
2004 FW	odd	projective, $f_6 = 0$	132M + 8S	120M + 12S
	odd	mixed, $f_6 = 0$	101M + 7S	
2005 FWW a	odd	projective, $f_6 = 0$	122M + 9S	110M + 11S
	odd	mixed, $f_6 = 0$	105M + 8S	
	even	projective, $H(x) = 1$, $f_6 = 0$	119M + 9S	42M + 15S
	even	mixed, $H(x) = 1$, $f_6 = 0$	102M + 8S	
2005 FWW b	even	$H(x) = 1$		I + 11M + 11S
	even	$H(x) = x$		I + 13M + 13S
	even	$H(x) = x^2$		I + 20M + 12S
	even	$H(x) = x^3$		I + 26M + 11S
2006 ATW	even	classical, $H(x) = 1$, $f_6 = 0$	I + 57M + 6S	I + 11M + 11S
	even	effective, $H(x) = 1$, $f_6 = 0$	I +47.7M + 6S	I + 9.3M + 11S

Table 2. Genus 3 hyperelliptic curve arithmetic – summary

Appendix C. Genus 4 hyperelliptic curves – summary

The following table compares various algorithms for genus 4 hyperelliptic curves. The hyperelliptic curve is of the form
$y^2+(h_4x^4+h_3x^3+h_2x^2+h_1x+h_0)y = x^9+f_8x^8+f_7x^7+f_6x^6+f_5x^5+f_4x^4+f_3x^3+f_2x^2+f_1x+f_0$.

Algorithm	char($\mathbb{k}$)	Properties	Addition	Doubling
1987 Cantor	general		6I + 386M/S	6I + 395M/S
2000 Nagao	odd	regular representation	3I + 286M/S	3I + 296M/S
	odd	alternative representation	2I + 292M/S	2I + 307M/S
2004 PWP	general	$h_i \in \{0, 1\}$, $f_8 = 0$	2I + 160M + 4S	2I + 193M + 16S
	even	$H(x) = x$, $f_8 = 0$	2I + 148M + 6S	2I + 75M + 14S
2006 ATW	even	classical, $H(x) = 1$,	I + 119M +10S	I + 28M + 16S
	even	effective, $H(x) = 1$,	I +98.1M +10S	I +23.7M + 16S

Table 3. Genus 4 hyperelliptic curve arithmetic – summary

Appendix D. Genus 3 C_{ab} Curves – Summary

The following table compares various algorithms C_{34} curves.

Algorithm	Curve	Algorithm Type	Addition	Doubling
1998 GPS	Superelliptic	LLL	20I + 600M	
1999 Arita	C_{ab}	Gröbner basis (Buchberger)		
2000 HS	C_{ab}	LLL		
2001 Bauer	$y^3 = F(x)$	LLL	10I + 547M	
2001 Arita	C_{34}	Explicit formulae	5I + 204M	5I + 284M
2002 BEFG	$y^3 = F(x)$	Cantor based	10I + 200M	
2003 FO	Picard	Explicit formulae	2I + 156M	2I + 174M
2003 BEFG	Picard	Explicit formulae	2I + 140M	2I + 164M
2003 BEFG	C_{34}	Explicit formulae	2I + 150M	2I + 174M
2004 FOR	Picard	Explicit formulae	2I + 130M	2I + 152M
2004 FOR	C_{34}	Explicit formulae	2I + 145M	2I + 167M
2007 ASM	C_{34}	Linear Algebra	2I + 117M	2I + 129M

Table 4. Genus 3 C_{ab} Curves Arithmetic – Summary

Appendix E. Implementation Results

Example E.0.2. The first example is a C_{23} (elliptic) curve over a prime field of size of the 250 bits prime number

$$783504955098126625939564619462229155911706009001203502697182381485670696613.$$

The elliptic curve is defined as $y^2 = x^3 + ax + b$, where

$$a = 67932267643457968166209555940584451919343347299360812115802973037931427095 7,$$

$$b = 77528910601603326472479335451945678736513769256954927837743210082920534314 7.$$

The base point P is set to be (x_0, y_0) where

$$x_0 = 66317733687373309421808144531088250138064976604969512753834018164547535747 0,$$

$$y_0 = 68975808239733787064730883358065000353293664399791187512477883059733111390 9.$$

The order of P is

$$1958762387745316564848911548655572889811124188417553782927142938666121046 75.$$

The example computes the scalar product mP where m is set to be

$$1958762387745316564848911548655572889811124188417553782927142938666121046 70.$$

Using Magma's built-in elliptic curves arithmetic the computation took 0.016 seconds, and gave the result (x_1, y_1) where

$$x_1 = 63019862982573127760531339108981032362362387592017401492469040142089174785 6,$$

$$y_1 = 12917565692266723499615524166679158798653555968872041531702467902052164851 8.$$

Using AMS algorithm the computation took 1.219 seconds, and returned the following Gröbner basis which corresponds to the ideal $\langle x - x_1, y - y_1 \rangle$.

$$\left\{ \begin{array}{l} y + 654329298175459390943409377795437567925170449312483087380157702465149048095, \\ x + 153306325272395348334251228372418832288082133081029487772491980064778948757 \end{array} \right\}$$

Example E.0.3. The following example is a C_{34} curve over the prime field of size 25033. The curve is defined to be

$$6567x^3y + y^3 + 25032x^4 + 18877x^2y + 162xy + 4738x^2 + 14333y + 7218x + 21234.$$

We represent the divisor $\mathrm{div}(x^3 + u_2x^2 + u_1x + u_0, y - (v_2x^2 + v_1x + v_0))$ by the sextuple $[u_2, u_1, u_0, v_2, v_1, v_0]$.

We select the divisor $D = [3904, 5539, 5752, 19670, 14925, 12954]$ and the scalar $m = 1341$ and compute mP.

Using FOR algorithm the computation took 0.016 seconds and returned the divisor $D_1 = [11095, 5932, 17083, 12380, 15154, 10043]$.

Using AMS algorithm the computation took 0.125 seconds and returned the following Gröbner basis

$$\left\{ \begin{array}{c} y^2 + 17380y + 4174x + 17473, \\ xy + 4646y + 21534x + 13556, \\ x^2 + 840y + 12437x + 14528 \end{array} \right\}$$

which corresponds to the ideal of D_1.

Note that BEFG algorithm is not applicable for this curve, because it is only effective for C_{34} curves where $h_3 = 0$.

Example E.0.4. The following example is a C_{34} curve over the prime field of size 2003. The curve is defined to be

$$y^3 + 2002x^4 + 1550x^2y + 1224xy + 1679x^2 + 856y + 1882x + 1302.$$

We select the divisor $D = [721, 1735, 1698, 1360, 1449, 1465]$ and the scalar $m = 10000$ and compute mP.

Using FOR algorithm the computation took 0.016 seconds and returned the divisor $D_1 = [1164, 1124, 904, 1260, 79, 581]$.

Using BEFG algorithm the computation took 0.016 seconds and returned the divisor $D_1 = [1164, 1124, 904, 1260, 79, 581]$.

Using AMS algorithm the computation took 0.125 seconds and returned the following Gröbner basis

$$\left\{ \begin{array}{c} y^2 + 258y + 921x + 279, \\ xy + 1683y + 50x + 1870, \\ x^2 + 1379y + 1224x + 1064 \end{array} \right\}$$

which corresponds to the ideal of D_1.

Example E.0.5. The following example is a C_{357} curve over the prime field of size 83. C is defined by

$$\left\{ \begin{array}{c} 64 + 4x + 30x^2 + 30x^3 + 76y + 75xy + y^2 + 52z + 4xz, \\ 10 + 27x + 16x^2 + 6x^3 + 44x^4 + 69y + 16xy + 27x^2y + 31z + xz + yz, \\ 22 + 32x + 77x^2 + 30x^3 + 11x^4 + 17y + 25xy + 3x^2y + 72x^3y + 45z + 32xz + 76x^2z + z^2 \end{array} \right\}$$

We select a point $P = (2, 33, 62)$. According to [AMS04] the order the ideal $I = \langle x - 2, y - 33, z - 62 \rangle$, corresponding to P, is $m = 1848$. We verified this statement. The computation of mI took 0.313 seconds and returned the Gröbner basis $\{1\}$.

Department of Mathematics, Bar-Ilan University, Ramat Gan, Israel
E-mail address: cohenran@yahoo.com