Implementation of imaginary quadratic fields and elliptic or hyperelliptic curves over finite prime fields on the system NZMATH for number theory

ANTONIO, Christine A. SAITO, Kentaro
TANAKA, Satoru ASUNCION, Janice S.
NAKAMULA, Ken
Department of Mathematics Tokyo Metropolitan University

Abstract

This paper is about the implementation of algorithms which determines the structure of 3 finite abelian groups on the system NZMATH for number theory. Namely, we want to compute the following: (i) the ring class groups of imaginary quadratic fields, (ii) the group of rational points of elliptic curves over $\mathbb{F}_p$, (iii) the jacobian of hyperelliptic curves of genus 2 over $\mathbb{F}_p$. All of these can be accomplished in a similar fashion. That is, for an arbitrary finite abelian group, if we can list up all the n elements of the group explicitly and execute the operation of two elements of the group, then we can determine the corresponding group structure. We may employ any of the three methods. First, we can make an operation table betweem the listed elements of the group. Unfortunately, this method is quite slow. Secondly, we can count the number of elements of prime power order for each prime factor of n, which requires the factorization of n and will be explained later. Lastly, we can apply the baby step giant step (BSGS) algorithm. This method is probabilistic but is the most efficient. For imaginary quadratic fields, we can employ the subexponetial algorithm under the extended Riemann hypothesis (ERH). For elliptic curves, we can use the known fact about the group structure and also the Weil pairing.

1 Computing the group structure of a finite abelian group

In order to compute the group structure of a finite abelian group of order n, we assume that all the n elements of the group are known explicitly and that the operation of two elements of the group is executable. First, we find the prime factorization

$$n = p_1^{n_1} \cdot p_2^{n_2} \cdots p_m^{n_m}.$$

If $n_1 = n_2 = \cdots = n_m = 1$, then the group structure is isomorphic to the cyclic group $\mathbb{Z}_n$. For the primes p_i of exponent greater than 1, we count the number of elements of order $p_i, p_i^2, \cdot\cdot\cdot, p_i^{n_i}$. The computations from the previous steps yields the desired group structure. In practice, there are not so many cases with many prime factors of n having high exponents.

2 Imaginary Quadratic Fields

2.1 Ring Class Groups of Imaginary Quadratic Fields

For $d \in \mathbb{Z}$, $d < 0$ and $d \equiv 0$ or $1 \pmod 4$, we have the following notations:

$O_d := \mathbb{Z} + \frac{d+\sqrt{d}}{2}\mathbb{Z}$: the imaginary quadratic order of discriminant d.

I_d: the multiplicative group of non-zero fractional O_d-ideals.

$I_d \geq P_d := \{\alpha O_d \mid 0 \neq \alpha \in \mathbb{Q}(\sqrt{d})\}$, the group of principal O_d-ideals.

$C_d := I_d/P_d$: the O_d-ideal class group, which is finite abelian.

If d is a fundamental discriminant, then C_d is the ideal class group of $\mathbb{Q}(\sqrt{d})$.

The main problem is to the determine the structure of C_d. This problem may be broken down into 3 steps.

2.2 The structure of C_d.

The first task is to list up the elements of C_d. Note that every element of C_d can be represented by a triple $(a, b, c) \in \mathbb{Z}^3$, such that

$$a > 0,\ b^2 - 4ac = d,\ \gcd(a, b, c) = 1.$$

The corresponding primitive O_d-ideal is

$$\mathrm{M}(a, b, c) = a\mathbb{Z} + \frac{-b + \sqrt{d}}{2}\mathbb{Z}.$$

We can make this representation unique by transforming to the reduced O_d-ideal such that

$$\frac{-b + \sqrt{d}}{2a} \in \mathcal{F}.$$

Here $\mathcal{F}$ is the set of $z \in \mathbb{C}$, $\mathrm{Im}(z) > 0$, satisfying either

$$|z| \geq 1 \text{ and } -\frac{1}{2} \leq \mathrm{Re}(z) \leq 0$$

or

$$|z| > 1 \text{ and } 0 < \mathrm{Re}(z) < \frac{1}{2}.$$

We have the following algorithm:

Algorithm 1 ***(Find the elements of*** C_d***.)***
INPUT: $d \in \mathbb{Z}$*,* $d < 0$ *and* $d \equiv 0$ *or* $1 \pmod 4$
OUTPUT: (a, b, c) *as above and the order of* C_d

This procedure is clear owing to the fact that

$$|b| \leq a \leq \sqrt{\frac{|d|}{3}}.$$

The second task is to determine the product of two primitive ideals. Its description is readily seen from the algorithm below:

Algorithm 2 ***(Multiplying primitive ideals)***
INPUT: Primitive O_d*-ideals* $M_1 = \mathrm{M}(a_1, b_1, c_1)$ *and* $M_2 = \mathrm{M}(a_2, b_2, c_2)$
OUTPUT: A primitive O_d*-ideal* $M = \mathrm{M}(a, b, c) \in M_1 M_2 P_d$

1. *Find* $s, t, u, v, w \in \mathbb{Z}$ *where*

$$s = \frac{b_1 + b_2}{2}, \ t = \gcd(a_1, a_2, s) = ua_1 + va_2 + ws.$$

2. *Set*

$$a = \frac{a_1 a_2}{t^2}, \ b = b_2 + \frac{2a_2}{t}(v(s - b_2) - wc_2), \ c = \frac{b^2 - d}{4a}.$$

Note however that the output here is not necessarily reduced. That part will be accomplished in the next algorithm.

The third task is to reduce primitive O_d-ideals which we obtained previously.

Algorithm 3 ***(Reduction of Ideals)***
INPUT: A primitive O_d*-ideal* $M_0 = \mathrm{M}(a_0, b_0, c_0)$
OUTPUT: The unique reduced primitive O_d*-ideal* $M = \mathrm{M}(a, b, c) \in M_0 P_d$

1. *Set* $(a, b, c) \leftarrow (a_0, b_0, c_0)$*.*

2. *If* $b \leq -a$ *or* $b > a$*, set* $b \leftarrow b - 2aq$*,* $c \leftarrow c + bq + aq^2$*, where* $b = 2aq + r$ *with* $-a < r \leq a$*.*

3. *If* $a > c$*, set* $(a, b, c) \leftarrow (c, -b, a)$ *and goto Step 1. Otherwise output* (a, b, c)*.*

With this, the operations in C_d may be executed explicitly.

Example 1 *Let* $d = -998775 = -3^2 \cdot 5^2 \cdot 23 \cdot 193$*. Then the* O_d*-ideal class group in* $\mathbb{Q}(\sqrt{-4439})$ *is given by* $C_d \cong \mathbb{Z} \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_{19}$*.*

3 Elliptic Curves

3.1 Preliminaries

Let $K = \mathbb{F}_q$ be a finite field with q elements and $\overline{K}$ be its algebraic closure.
For $g \in \mathbb{Z}$, $g > 0$, $h(u)$, $f(u) \in K[u]$, $\deg h \leq g$, $\deg f = 2g + 1$ and f is monic, we will use the following notations:

C: (non-singular) hyperelliptic curve of genus g defined over K by

$$C : v^2 + h(u)v = f(u).$$

$\mathbb{D}$: the (additive) group of divisors of C.

$\mathbb{D}^0$: the group of divisors of degree 0.

$\mathbb{P}$: the group of principal divisors $\mathrm{div}(R)$ for some non-zero rational function $R \in \overline{K}(C)$. For D_1, $D_2 \in \mathbb{D}$, we denote $D_1 \sim D_2$ if $D_1 - D_2 \in \mathbb{P}$.

$\mathbb{J} = \mathbb{D}^0/\mathbb{P}$: the jacobian of C.

$\mathbb{J}(K)$: the group of all divisor classes in $\mathbb{J}$ that have a representative that is defined over K. This is finite abelian.

The main problem is to determine the structure of $\mathbb{J}(K)$.

A hyperelliptic curve of genus 1 is an *elliptic curve*, and we denote $E = C$.
In this case, the group $E(K)$ of K rational points of E is isomorphic to $\mathbb{J}(K)$.

In 2004, Komai implemented `NZMATH.elliptic` the basic package for elliptic curves on NZMATH [3]. For this project, we extend the package by including Weil-Pairing and Miller's algorithm which computes the group structure of an elliptic curve over $K = \mathbb{F}_p$ for a finite prime field.

3.2 The Weil Pairing

The points of finite order on an elliptic curve are two-dimensional. More specifically, suppose that K is a field of characteristic p, which is perfect(i.e., $K^p = K$), Ω is a fixed algebraic closure of K, and n is a positive integer, relatively prime to p. For $P \in E$, we denote its divisor by $(P) \in \mathbb{D}$ and $E[n]$ as the set of points on E of order dividing n. As an Abelian Group, we have

$$E[n](\Omega) \cong \mathbb{Z}_n \times \mathbb{Z}_n$$

where $\mathbb{Z}_n$ denotes the cyclic group of order n.

The Weil pairing is a non-degenerate inner product defined on points of $E[n](\Omega)$. Unlike the more familiar inner product defined on vector spaces, it is alternating (i.e. for all v we have $\langle v, v \rangle = 0$).

For a divisor $D = \sum_{P \in E} m_P(P) \in \mathbb{D}$, the support of D is given by

$$\{P \in E | m_P \neq 0\}.$$

Definition 1 *(Weil Pairing) Let $n > 1$ be an integer, $P,\ Q \in E[n]$, and let $D_P \sim (P) - (\mathcal{O})$, $D_Q \sim (Q) - (\mathcal{O})$ be divisors on an elliptic curve, E, with disjoint supports, such that $nD_P, nD_Q \sim 0$. This means that there are functions f_P and f_Q such that $nD_P = \operatorname{div}(f_P), nD_Q = \operatorname{div}(f_Q)$. We define the Weil pairing by*

$$e_n(P, Q) = \frac{f_P(D_Q)}{f_Q(D_P)}.$$

To compute $e_m(P, Q)$, we pick random points $R_1, R_2 \in E(K)$ such that the divisor $D_P = (P + R_1) - (R_1)$ and $D_Q = (Q + R_2) - (R_2)$ have disjoint support. Then

$$e_n(P, Q) = \frac{f_{D_P}((Q + R_2) - (R_2))}{f_{D_Q}((P + R_1) - (R_1))} = \frac{f_{D_P}(Q + R_2)}{f_{D_P}(R_2)} \frac{f_{D_Q}(R_1)}{f_{D_Q}(P + R_1)}.$$

Miller's algorithm computes $f_{D_P}(Q),\ Q \neq \mathcal{O}$.

Input: $K,\ E/K,\ P, Q \in E[n]\ n = \sum_{i=0}^{t} k_i 2^i,\ k_i \in \{0, 1\}$

Output: $f_{D_P}(Q)$

We do not go into the details of this procedure. Please refer to [2].

3.3 Weil Pairing with Miller's Algorithm

We now give a short description of an algorithm by Miller:

Algorithm 4 ***(Miller)***
INPUT: $K,\ E/K,\ P, Q \in E[n]$
OUTPUT: $e_n(P, Q)$

1. *Find Point $R_1, R_2 \in E[n]$.*

2. *Set $f = \frac{f_{D_P}(Q+R_2)}{f_{D_P}(R_2)} \frac{f_{D_Q}(R_1)}{f_{D_Q}(P+R_1)}$.*

3. *Return f.*

This algorithm has a very high probability of working with the choice of R_1, R_2 (the probability of failure is at most $O(\frac{\log p}{p})$).

In the rare event that a division by zero occurs during the computation of $e_n(P, Q)$ we simply pick new random points R_1, R_2 and repeat the process.

3.4 The Group Structure of $E(K)$

In this subsection we discuss the group structure of the group $E(K)$ where K is a finite field. In particular we give a fast algorithm for determining the group structure which makes essential use of the fast algorithm to calculate the Weil pairing.

It is known that the group structure of $E(K)$ is always either cyclic, or the product of two cyclic group order d_1 and d_2 where $d_1 \mid d_2$.

The following algorithm runs in polynomial time $(\log q)$ if the factorization of $\gcd(q-1, N)$ is known, where N is the order of the $E(K)$.

Algorithm 5 ***(Finding the group structure of*** $E(K)$***)***
INPUT: K*,* E/K
OUTPUT: The group structure (n_1, n_2) *of* $E(K)$

```
1. Compute exact $N = \#E(K)$.
2. Compute $N = N_0 N_1$ where $\gcd(N_0, N_1) = 1$, and the set of prime divisors of $N_0$
     is the same as the set of prime divisors of gcd$(q-1, N)$.
3. Take two points $P', Q' \in E(K)$, and set $P = N_1 P', Q = N_1 Q'$.
4. Find the exact order of $P$ and $Q$ (this is where we need the factorization of $N_0$).
     Say they are $m$ and $n$.
5. Set $r = \operatorname{lcm}(m, n)$.
6. Set $a = e_r(P, Q)$.
7. Find the exact order of $a$. Say it's $s$.
8. If $rs = N_0$:
     Return $(rN_1, s)$.
     Else:
     Go to step 3.
```

Example 2 $E : y^2 = x^3 + 4$ *over* F_{997}
$\#E(F_{997}) = 1008$
$E(F_{997}) \cong \mathbb{Z}_{12} \times \mathbb{Z}_{84}$

4 Hyperelliptic Curves

A hyperelliptic curve of genus greater than 1 has a more complicated arithmetic than elliptic curves since the rational points over K do not form a group.

In our NZMATH package, we implemented algorithms to compute the group structure of the jacobian $\mathbb{J}(K)$ of hyperelliptic curves of genus $g = 2$ over finite prime fields $K = \mathbb{F}_p$.

4.1 Mumford Representation

Every $D \in \mathbb{J}(K)$ is represented by a semi-reduced divisor. For a detailed discussion, please refer to [5]. For any polynomials $a(u)(\neq 0)$, $b(u) \in \overline{K}[u]$, we denote $\mathrm{div}(a, b) = \gcd(\mathrm{div}(a(u)), \mathrm{div}(b(u) - v))$.
Further, every $D \in \mathbb{J}(K)$ can be uniquely represented by two polynomials $a(u), b(u) \in K[u]$ so that $D = \mathrm{div}(a, b)$, which is called a reduced divisor, where

1. $a(u)$ is monic,
2. $\deg_u b < deg_u a \leq g$, and
3. $a(u)$ divides $(b(u)^2 + b(u)h(u) - f(u))$.

In the case of genus $g = 2$ hyperelliptic curve, we can easily list up all the reduced divisors.

4.2 Adding Reduced Divisors

Now we will present an algorithm on how to effectively add 2 elements of the jacobian, known as **Cantor's Algorithm**.

Algorithm 6 ***(Cantor)***
INPUT: Reduced divisors $D_1 = \mathrm{div}(a_1,b_1)$ *and* $D_2 = \mathrm{div}(a_2,b_2)$
OUTPUT: A semi-reduced divisor $D = \mathrm{div}(a,b)$ *defined over* K *such that* $D \sim D_1 + D_2$

1. *Use the extended Euclidean algorithm to find polynomials* d_1, e_1, $e_2 \in K[u]$ *where* $d_1 = \gcd(a_1,a_2)$ *and* $d_1 = e_1a_1 + e_2a_2$.

2. *Use the extended Euclidean algorithm to find polynomials* d, c_1, $c_2 \in K[u]$ *where* $d = \gcd(d_1, b_1 + b_2 + h)$ *and* $d = c_1d_1 + c_2(b_1 + b_2 + h)$.

3. *Let* $s_1 = c_1e_1$, $s_2 = c_1e_2$, *and* $s_3 = c_2$, *so that*

$$d = s_1a_1 + s_2a_2 + s_3(b_1 + b_2 + h)$$

Set

$$a = \frac{a_1a_2}{d^2}$$

$$b = \frac{s_1a_1b_2 + s_2a_2b_1 + s_3(b_1b_2 + f)}{d} \bmod a.$$

Unfortunately, this algorithm yields only a semi-reduced divisor. Next, we present an algorithm that finds its equivalent reduced divisor.

Algorithm 7 ***(Computing reduced divisors)***
INPUT: A semi-reduced divisors $D = \mathrm{div}(a,b)$ *over* K.
OUTPUT: A unique reduced divisor $D'=\mathrm{div}(a',b')$ *such that* $D' \sim D$.

```
1. Set
       $a' = \frac{f - bh - b^2}{a}$
   and
       $b' = -h - b(\mathrm{mod} a')$
2. If $\deg_u a' > g$ then set $a \leftarrow a'$, $b \leftarrow b'$ and go to step 1.
3. Let $c$ be the leading coefficient of $a'$, and set $a' \leftarrow c^{-1}a'$.
4. Output $(a',b')$.
```

Example 3 $C : v^2 + uv = u^5 + 5u^4 + 6u^2 + u + 3$
$F_p = F_{227}$
number of points in $C(F_p)$*: 183*
number of special point(s): 1
number of reduced divisors: 42212
Group Order: 42396=$2^3 \cdot 3 \cdot 3533$
Div of Order 2: 3
Since there are three elements of order 2, then $J(F_p)$ *over* F_{227} *is isomorphic to* $\mathbb{Z}_2 \times \mathbb{Z}_{21198}$.

5 Future Implementations

- For the imaginary quadratic field part, we are currently working on a program to compute the group structure of the ring class group using the BSGS algorithm. Next, we plan to implement the subexponetial algorithm under the ERH.

- For elliptic curves, the BSGS to determine the group structure of its rational points is also in progress. We also intend to use the Weil pairing for cryptologic application.

- For the hyperelliptic package, since the program to compute the group structure is rather slow, we are working on the BSGS program (from the paper of Matsuo, et.al. entitled *Baby Step Giant Step Algorithm for Point-Counting on Hyperelliptic Curves*) to make the computation faster.

References

[1] H. Cohen.: A Course in Computational Algebraic Number Theory *Springer-Verlag.* (1993).

[2] V.Miller.: The Weil Pairing, and Its Efficient Calculation *J. Crypto.* **17**. (2004), 235-261.

[3] Hiroaki Komai: Implementation of Arithmetic of Elliptic Curves over Finite Prime Fields on NZMATH, http://tnt.math.metro-u.ac.jp/labo/master/2004/komai/syuron_e.ps .

[4] M. Jacobson Jr., A. Menezes, A. Stein.: Hyperelliptic Curves and Cryptography http://www.cacr.math.uwaterloo.cal/ajmenezes/publications/hcc.pdf .

[5] A. Menezes, Y. Wu, R. Zuccherato.: An Elementary Introduction to Hyperelliptic Curves *Technical Report CORR 96-19.* (1996) .

[6] J. Silverman.: The Arithmetic of Elliptic Curves *Springer-Verlag.* (1986).