The Pennsylvania State University
The Graduate School

ALGORITHMS FOR ABELIAN SURFACES OVER FINITE FIELDS AND THEIR APPLICATIONS TO CRYPTOGRAPHY

A Dissertation in
Mathematics
by
Hao-Wei Chu

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

August 2021

The dissertation of Hao-Wei Chu was reviewed and approved by the following:

Kirsten Eisenträger
Professor of Mathematics
Francis R. Pentz and Helen M. Pentz Professor of Science
Dissertation Advisor
Chair of Committee

Jack Huizenga
Associate Professor of Mathematics

Wen-Ching Winnie Li
Distinguished Professor of Mathematics

Martin Fürer
Professor of Computer Science and Engineering

Alexei Novikov
Professor of Mathematics
Director of Graduate Studies

Abstract

This dissertation investigates two types of abelian surfaces: superspecial abelian surfaces over finite fields and abelian surfaces over number fields with complex multiplication. We generalize theorems for elliptic curves to these surfaces, and discuss their applications in cryptography.

In the first part, by extending Page's algorithm in 2014, we give a probabilistic algorithm that solves principal ideal problems over matrix algebras over quaternion algebras in subexponential time in the size of the ideal and the determinant of the quaternion algebra. We also discuss their applications to cryptography protocols based on isogenies on superspecial abelian surfaces.

In the second part, we discuss a p-adic algorithm which computes the Igusa class polynomial of a quartic CM field, which encodes abelian surfaces with complex multiplication by the field. We discuss potential improvements to the canonical lifting algorithm by Carls and Lubicz in 2009, which is the core of the p-adic algorithm. Applying the improvement, we computed examples for $p = 5$ and 7. We also analyze the computational complexity for the entire p-adic algorithm.

Table of Contents

Chapter 3

Appendix

List of Figures

Acknowledgements

First, I would like to offer my deepest appreciation and respect to my thesis advisor, Professor Kirsten Eisenträger, for all her patient guidance throughout my Ph. D. study, for her challenges when I got frivolous, for her empowerment when I faltered, and for all her care during the pandemic, when all the world just collapsed in a sudden. Without her help, I might not stay sane, let alone starting a thesis.

Second, I would like to thank the committee members of this dissertation, including Professor Winnie Wen-Ching Li, Professor Jach Huizenga, and Professor Martin Fürer, for all their input and challenges to the dissertation and the defenses. I would also like to thank the mathematics department staff, Allyson Borger, for keeping everything handled and seamlessly smooth, including directing me to the LaTeXtemplate of this thesis, so that I can keep our concern outside our study minimal.

Next, I would like to thank my academy brother, Caleb Springer, for organizing the study groups in the math department, and for being a good role model as a scholar. I would also like to thank Chien-Hua Chen and other number theory graduate students and all my roommates and friends, for illuminating my life at Penn State.

Also, I would like to thank everyone who taught me math. In particular, I thank Professor Jing Yu, for showing me both his affinity and self-discipline while being one of the leading mathematicians in Taiwan, and Professor Chia-Fu Yu, for showing me the humbleness and hardworking, and for directing me to supersingular abelian varieties.

Finally, I would like to thank my parents, for maintaining a cozy home while supporting me to pursue my dream; and my little sister, for all the sweet conversations and nice pieces of career advice, and for taking care of my family while I am not at home.

Hao-Wei Chu is partially supported by National Science Foundation grants CNS-1617802 and CNS-2001470 during his Ph. D. program at Penn State and the writing of the dissertation. The findings and views of this dissertation do not necessarily reflect the views of the National Science Foundation.

Chapter 1
Introduction

In the thesis, we will be giving algorithms for computational problems arising from abelian surfaces, and discuss the connections between these computational problems and cryptography.

Elliptic curves over finite fields play an important role in public-key cryptography. There are two types of elliptic curves, ordinary elliptic curves, and supersingular elliptic curves. We can characterize these two types using their endomorphism rings $\mathrm{End}_{\overline{\mathbb{F}_p}}(E)$ over the algebraic closure: The endomorphism ring of an ordinary elliptic curve is an order of an imaginary quadratic field; while the endomorphism ring of a supersingular elliptic curve is an order in a definite quaternion algebra. It is also possible to classify them according to their p-torsion, denoted $E[p]$.

Both ordinary and supersingular elliptic curves have been studied in the public-cryptographic system: The classical computational hardness of the discrete log problem on ordinary elliptic curves has been the keystone of the elliptic curve Diffie-Hellman (ECDH) and many other protocols. On the other hand, there are several problems related to supersingular elliptic curves which are believed to be computationally hard even assuming the quantum computers, such as finding isogenies between supersingular curves of prime power degree and computing the endomorphism ring of a general supersingular curve. For instance, the SIDH algorithm in [DJP14] relies on the isogeny problem, and the algorithm proposed by Galbraith et al. in [GPS19] used the endomorphism problem of supersingular elliptic curves to construct a signature scheme.

It is natural to ask for a higher-dimensional analog of isogeny-based cryptography, which stimulated numerous analogous questions to be asked on higher-dimensional abelian varieties. In this thesis, we will be discussing abelian varieties of dimension $g = 2$, and in this case, every simple abelian variety is geometrically isomorphic to the Jacobian variety of a hyperelliptic curve. For abelian varieties of dimension > 1, it is no longer true that they are either ordinary or supersingular. We call an abelian variety A of dimension g ordinary, if $A[p] = (\mathbb{Z}/p\mathbb{Z})^g$. We call A supersingular if A is geometrically isogenous to a product of supersingular elliptic curves, and a supersingular abelian variety A is said to be superspecial if it is isomorphic to a product of supersingular elliptic curves.

This thesis investigates the generalization of cryptographic primitives to abelian surfaces for both the ordinary and superspecial cases. After a brief overview of the classifications and endomorphism rings of abelian surfaces, We will focus on algorithms that are essential to these generalizations. In Chapter 3, we will describe and analyze an algorithm for the principal ideal problem over a central simple algebra. We will also discuss why the principal ideal problem plays an important role in generalizing the signature scheme proposed by Galbraith et al. in [GPS19, Chapter 4]. In Chapter 4, we will discuss improvements and implementation aspects of the p-adic method in computing the Igusa class polynomials, the dimension 2 analog of Hilbert class polynomials.

1.1 Isogeny-Based Cryptosystems on Abelian Surfaces

In public-key cryptography, most protocols depend on some underlying computationally hard problem. In the case of protocols using supersingular elliptic curves, the underlying problem is typically one of the following:

- (Finding isogeny path) Let E, E' be two supersingular elliptic curves over a finite field $\mathbb{F}_q$, find an isogeny from E to E' which has a smooth degree, such as the hash function in [CGL08] and the Diffie-Hellman key exchange scheme in [DJP14].

- (Computing the endomorphism ring) Let E be a supersingular elliptic curve over a finite field. Compute a $\mathbb{Z}$-basis of $\mathrm{End}(E)$.

We are particularly interested in protocols involving supersingular elliptic curves and their endomorphism rings, such as the signature scheme by Galbraith et al. in [GPS19].

For a supersingular elliptic curve E_0 over $\overline{\mathbb{F}_p}$, we know that E_0 has a model over $\mathbb{F}_{p^2}$. Furthermore, take $\mathcal{O}_0 := \mathrm{End}_{\overline{\mathbb{F}_p}(E_0)}$. Then $\mathcal{O}_0$ is a maximal order of the quaternion algebra $B_{p,\infty}$, where $[B_{p,\infty} : \mathbb{Q}] = 4$ and is ramified exactly at p and infinity.

Deuring's correspondence (see [Voi20, Theorem 42.3.2] for details) gave a bridge between isomorphism classes of such E_0 and objects in quaternion algebras. Suppose we fix a "base supersingular curve" E_0 over $\overline{\mathbb{F}_p}$, and take $\mathcal{O}_0$ and $B_{p,\infty}$ as above, then the mapping

$E \mapsto \mathrm{Hom}(E, E_0)$ defines a contravariant functor between the following categories:

$$\left\{\begin{array}{c}\text{Supersingular elliptic curves over } \overline{\mathbb{F}_p},\\ \text{morphisms are } \overline{\mathbb{F}_p}\text{-isogenies}\end{array}\right\} \to \left\{\begin{array}{c}\text{Invertible left } \mathcal{O}_0\text{-modules, morphisms}\\ \text{are left } \mathcal{O}_0\text{-module homomorphisms}\end{array}\right\}$$

Composing the map $I \to \mathcal{O}_R(I)$ with the map above, we obtain a bijection between $\mathrm{Gal}(\overline{\mathbb{F}_p}/\mathbb{F}_p)$-orbits of the j-invariants of supersingular elliptic curves over $\overline{\mathbb{F}_p}$ and conjugacy classes of maximal orders in $B_{p,\infty}$. The correspondence also provided us a pathway between isogeny paths and one-sided ideals in $B_{p,\infty}$. For instance, [KLPT14] proposed an algorithm to find ideals with norms of the form ℓ^k for a small prime ℓ in one-sided ideal classes in $B_{p,\infty}$, which can be applied to find ℓ-isogeny paths when some knowledge regarding the endomorphism ring is given.

In the 2-dimensional case, much less is known. There is a hash function ([Tak18] and [CDS20]) and a Diffie-Hellman key exchange algorithm ([FT19]) published recently. As we will demonstrate in Chapter 2, the closest analogy for supersingular elliptic curves over $\overline{\mathbb{F}_p}$ are superspecial abelian varieties over $\overline{\mathbb{F}_p}$.

Given a prime p, we will fix a supersingular elliptic curve E defined over $\mathbb{F}_p$ and an embedding $\mathrm{End}(E) \hookrightarrow \mathcal{O} \subsetneq B_{p,\infty}$. It is possible to identify every superspecial abelian surface with polarization $(E^2, \mathscr{L})$ as a conjugate $g\Lambda g^{-1}$, for some $g \in \mathrm{Mat}_2(\mathcal{O})$. And this gives us an analogy to the endomorphism rings of supersingular elliptic curves.

To generalize the signature scheme in [GPS19] into superspecial abelian surfaces, one of the most important ingredients needed is an analogy for the following fact: Although finding endomorphism rings of supersingular elliptic curves is hard in general, in the special case that an isogeny $E_0 \to E_1$ is given and $\mathcal{O}_0 = \mathrm{End}(E_0)$ is known, $\mathcal{O}_1 = \mathrm{End}(E_1)$ can be computed efficiently.

When generalizing this to higher dimension, it turned out that the principal ideal problem is essential: Let $B_{p,\infty}$ be the quaternion algebra and $\mathcal{O}$ be a maximal order of $B_{p,\infty}$ as before. Suppose $I \subseteq \mathrm{Mat}_2(B_{p,\infty})$ is a right $\mathrm{Mat}_2(\mathcal{O})$-order (which is known to be always principal). How do we find a right ideal generator α of I?

We will discuss the principal ideal problem over the matrix ring over a quaternion algebra over $\mathbb{Q}$, with the idea based on [Pag14]. And our main contribution is to provide an algorithm and analyze the complexity which yields the following theorem:

Theorem A. *Suppose $B_{p,\infty}$ is a quaternion algebra over $\mathbb{Q}$, ramified exactly at p and ∞ for some prime p, $\mathcal{O}$ be a maximal order in $B_{p,\infty}$. Let I be a right* $\mathrm{Mat}_2(\mathcal{O})$ *order.*

Then, under certain heuristic assumptions, Algorithm 2.3.1 gives a probabilistic algorithm that runs in subexponential time in p and finds the principal ideal generator of I.

1.2 Finding CM Abelian Surfaces via the p-Adic Approach

In the elliptic case, to obtain optimal security for the discrete log problem, we need to find a curve E over a finite field $\mathbb{F}_q$ such that the number of points, $|E(\mathbb{F}_q)|$ is a large prime or contains a large prime factor. In practice, there are two approaches: (1) By generating a random curve E, computing the number of points, and repeat until a curve with suitable order is found; or (2) Start with a suitable imaginary quadratic field K, and use the theory of complex multiplication (CM) to find such a curve. We will consider the second approach.

Hilbert class polynomials play an important role in the classical CM theory. For an imaginary quadratic field K, the Hilbert class polynomial is a monic polynomial whose roots are exactly the j-invariants of elliptic curves with CM by K, i.e. the endomorphism ring is $\mathcal{O}_K$. The Hilbert class polynomial has integral coefficients, and its splitting field is the Hilbert class field of K.

For the genus 2 case, it turns out that the j-invariants of elliptic curves with CM by a quadratic field can be replaced by the Igusa invariants (i_1, i_2, i_3) of hyperelliptic Jacobians with CM by a quartic field. As we will see, the associated Igusa polynomials still enjoy some of the key properties of Hilbert class polynomials. In particular, one can recover from the Igusa polynomials hyperelliptic Jacobians with CM by a quartic number field K and one can construct hyperelliptic curves for use in discrete logarithm problems.

There are three major approaches to construct Igusa class polynomials: (1) the analytic approach [Str14, ET14]; (2) the CRT approach [EL10]; and (3) the p-adic approach (see [GHK$^+$06] for $p = 2$; and [CKL08, CL09] for $p = 3$). Our main goal in Chapter 4 is to investigate the possibilities of the p-adic approach.

The p-adic method for computing the Igusa class polynomial proceeds as follows: (1) Search for an abelian surface A over a finite field $\mathbb{F}_{p^r}$ such that $\mathrm{End}(A) \cong \mathcal{O}_K$; (2) Find an abelian surface $\tilde{A}$ over $\mathbb{Q}_{p^{r'}}$, the degree r' unramified extension of $\mathbb{Q}_p$, such that $\mathrm{End}(\tilde{A}) \cong \mathcal{O}_K$; (3) Recover the Igusa invariants and Igusa polynomials from $\tilde{A}$.

The core of the p-adic approach is step (2) above, for which we will take $\tilde{A}$ as the canonical lift of A (see Definition 3.2.10 for details). Since the canonical lift also lifts the Frobenius, this leads to certain conditions that must be satisfied by the canonical lift. In [GHK$^+$06], Gaudry et al. gave an approach for $p = 2$ using Rosenhein invariants. For $p = 3$, [CKL08] proposed a method using the 4-theta null points, but their method does not generalize. In [CL09], a method for general p and any dimension g is proposed using $2^n p$-theta null points, but it turned out to be computationally feasible only when $p = 3$. Faugére et al. discussed improvements of the Gröbner basis step in [DJP14] and gave experimental results to compute $2p$-theta null points for $p = 3$ or 5.

In Chapter 3, we will make improvements to the p-adic algorithms to compute Igusa class polynomials and give a complexity analysis of the algorithm. The complete statements of the theorem will be in Theorems 3.1.1 and 3.1.2. A simplified statement of the canonical lifting (which is the key step of the algorithm) and the overall algorithm complexity can be given as follows:

Theorem B. *Let K be a quartic CM field of discriminant D. Suppose C is a hyperelliptic curve of genus 2, defined over some finite field $\mathbb{F}_q = \mathbb{F}_{p^r}$, such that the endomorphism ring of $\mathrm{Jac}(C)$ is the maximal order $\mathcal{O}_K$ of K. Further, assume that the $2p$-theta null points of $\mathrm{Jac}(C)$ are defined over $\mathbb{F}_{q^d}$. Then the canonical lifting of $\mathrm{Jac}(C)$ can be computed in $\tilde{O}(p^{24p+32} + D^3)$ operations in the finite field $\mathbb{F}_q$.*

Moreover, assume that X represents the time required to determine whether a Jacobian over $\mathbb{F}_q$ has CM by K. Then, using the p-adic approach, computing the Igusa polynomial takes time complexity

$$\tilde{O}\left(\frac{q^3 + q^{3/2}X}{\sqrt[4]{D}}\right) + \tilde{O}\left(p^{24p+32} + D^3\right) + \tilde{O}(D^5).$$

An upper bound for X can be found in [FL08].

We also implemented our algorithm and provide examples of 5-adic and 7-adic methods. To the knowledge of the author, only 2-adic and 3-adic examples are given in the existing literature.

Chapter 2
Principal Ideal Generator Problems over Matrix Rings of Quaternion Algebras

2.1 Introduction

2.1.1 Isogeny-based cryptosystems

Public-key cryptography have gained increasing and irreplaceable importance in establishing and managing secure communication since RSA was published in late 1970's. The majority of such public-key cryptosystems have their security relying on some hard mathematical problems, and the most widely used protocols relied on the hardness of integer factorization and discrete logarithm problem over a specific group.

As a consequence of the invention of some quantum algorithms which undermines the hardness of factorization and discrete logarithm problems, finding replacements for the current public key cryptosystems has became a crucial task. In 2016, NIST initiated a project (announced at [Nat16]) to call for proposals of quantum-resilient protocols, and aimed for their evaluation and standardization.

Isogeny based cryptosystem had been one of the proposals sent to [Nat16]. The key ingredient of the first isogeny based cryptography systems is the hardness of the following problems regarding isogenies of supersingular elliptic curves.

- (Finding isogeny path) Let E, E' be two supersingular elliptic curves over a finite field $\mathbb{F}_q$, find an isogeny from E to E' satisfying certain condition.

- (Computing the endomorphism ring) Let E be a supersingular elliptic curve over a finite field. Compute a $\mathbb{Z}$-basis of $\mathrm{End}(E)$.

For instance, the first public key protocol which used isogenies on supersingular elliptic curves can be dated back to the hash function proposed by Charles, Goren and Lauter in [CGL08], which used the fact that the ℓ-isogeny graph over isomorphism classes of supersingular ellipic curves over $\overline{\mathbb{F}_p}$ is an expander graph. And later in 2011, de Feo, Jao and Plût proposed in [DJP14] the SIDH algorithm, which is a variant of Diffie-Hellman, using 2 and 3-isogeny of supersingular elliptic curves over $\overline{\mathbb{F}_p}$, and the SIDH algorithm were developed, packaged into the SIKE algorithm proposed to the NIST post-quantum algorithm challenge. More recently, Castrtck et al. proposed CSIDH, which utilized the isogeny grapgs on supersingular elliptic curves over the prime field $\mathbb{F}_p$.

An alternative approach of using supersingular elliptic curves in crpytographic algorithms is to use the hardness of computing endomorphism rings of a supersingular elliptic curve. For instance, Galbraith et al. proposed in [GPS19] a signature algorithm which utilizes both the hardness of computing endomorphism ring of an arbitrary supersingular elliptic curve over the finite field $\mathbb{F}_q$ and the hardness of finding an isogeny path.

It is well known that supersingular elliptic curves and their isogeny graphs are deeply connected to orders and quaternion algebras. For any prime p, we know that the endomorphism ring of a supersingular elliptic curve over $\overline{\mathbb{F}_p}$ is a maximal order in $B_{p,\infty}$, the quaternion algebral over $\mathbb{Q}$ which ramifies exactly at p and ∞. In addition, the Deuring correspondence gives a bijective map between isomorphism classes of supersingular elliptic curves over $\overline{\mathbb{F}_p}$ and equivalence classes of maximal orders in $B_{p,\infty}$. Finding the Deuring correspondence for a supersingular elliptic curve is in general hard and equivalent to computing the endomorphism and finding isogeny path, as shown in [EHL$^+$18], but when the correspondence is provided, one can often transform problems on supersingular elliptic curves to problems on quaternion orders. For instance, suppose that the supersingular elliptic curves E, E' are over $\overline{\mathbb{F}_p}$ and correspond to the maximal orders $\mathcal{O}$ and $\mathcal{O}'$, respectively. Then finding an isogeny $E \to E'$ of powersmooth degree can be reduced to finding an element of powersmooth norm in I, where I is a connecting ideal of $\mathcal{O}$ and $\mathcal{O}'$. This element can be found efficiently using a modified version of the algorithm of Kohel et al. in [KLPT14].

Generalizing isogeny-based cryptographic algorithms to higher dimensional abelian varieties has also gained more interest recently. The first idea was to use principally polarized supersingular abelian surfaces which are Jacobians of hyperelliptic curves over $\overline{\mathbb{F}_p}$, and the first dimension 2 algorithm was known to be a hash algorithm proposed by Takashima in [Tak18]. Flynn and Ti found in [FT19] a weakness in Takashima's algorithm by showing the existence of short cycles and instead proposed a Diffie-Hellman protocol on the $(2,2)$- and $(3,3)$-isogeny graphs on supersingular abelian surfaces, analogous to SIDH. In contrast, instead of supersingular abelian surfaces (those which are isogenous to a product of supersingular elliptic curve), Castryck and Smith narrowed the object to superspecial (those which are isomorphic to a product of supersingular elliptic curves) hyperelliptic Jacobians. They proposed in [CDS20] a modified hash function, which is claimed to be immune short cycles collision attacks (compared to Takashima's proposal [Tak18]).

2.1.2 Superspecial abelian surfaces and matrix rings over quaternion algebras

It seems that in dimension 2, superspecial hyperelliptic Jacobians of genus 2 shared more common properties with supersingular elliptic curves. Just to mention a few, every superspecial hyperellipic Jacobian over $\overline{\mathbb{F}_p}$ can be defined over $\mathbb{F}_p$ or $\mathbb{F}_{p^2}$, and the endomorphism ring is a maximal order in $\mathrm{Mat}_2(B_{p,\infty})$. Since $\mathrm{Mat}_2(B_{p,\infty})$ satisfy the Eichler's criterion, it follows from strong approximation that all maximal orders in $\mathrm{Mat}_2(B_{p,\infty})$ are conjugate to each other, and the class number of the maximal orders are 1 Therefore, the class number does not enumerate the hyperelliptic curves; instead, as described by Ibukiyama et al. in [IKO86], taking $\mathcal{O}$ to be a maximal order in $B_{p,\infty}$, one can consider $\mathcal{O}$-lattices of rank 2, and define an equivalence relation on the lattices. The number of principal polarizations on such superspecial abelian surfaces is then bijective to the lattice classes. Ibukiyama et al. gave an explicit formula to compute the number of principal polarizations of a superspecial abelian surface and the number of superspecial hyperelliptic Jacobians in [IKO86, Theorem 3.1, 3.3] (see also Brock's thesis in [Bro93, Theorem 3.10A]). This is a generalization to the mass formula in the quaternion algebra case.

Knowing the relations between superspecial abelian varieties, orders in $\mathrm{Mat}_2(B_{p,\infty})$, and $\mathcal{O}$-lattices of rank 2, we are interested in the questions which translates a superspecial hyperelliptic Jacobian of genus two to maximal orders, converts a computational problem in hyperelliptic to a relative problem in $\mathrm{Mat}_2(M_{p,\infty})$, or vice versa. Generalizing Galbraith's isogeny based signature algorithm is a standard example which involves many aspects of the ingradient: We need to know how to convert an (ℓ,ℓ)-isogeny between superspecial hyperelliptic Jacobians of genus two to an element in the central simple algebra $\mathrm{Mat}_2(B_{p,\infty})$ and vice versa; and we also need a $\mathrm{Mat}_2(B_{p,\infty})$ analog of the powersmooth element algorithm as proposed by Kohel et al. in [KLPT14]. Finding an (ℓ,ℓ)-isogeny with kernel represented in the $\mathrm{Mat}_2(B_{p,\infty})$ side is one of the problems involved in the generalization process, and an algorithm for the following question turned out to be critical:

Problem 2.1.1. Let $B_{p,\infty}$ be the quaternion algebra over $\mathbb{Q}$, ramifying at some prime p and infinity, and let $\mathcal{O}$ be a maximal order in $B_{p,\infty}$, and let ℓ be a prime different from p. Given

$\tilde{\gamma} \in \mathrm{Mat}_2(\mathcal{O}/\ell\mathcal{O})$, find $\gamma \in \tilde{\gamma} + \ell\mathrm{Mat}_2(\mathcal{O})$ satisfying $N(\gamma) = \ell^2$.

And we can answer Problem 2.1.1 if there exists an efficient algorithm for the principal ideal problem over matrix algebras over the indefinite quaternion algebra $B_{p,\infty}$:

Problem 2.1.2 (Principal ideal problem)**.** Let $B_{p,\infty}$ be the quaternion algebra over $\mathbb{Q}$, ramifying at p and the infinite place, and $\mathcal{O}$ be a maximal order of B. Let I be a right $\mathrm{Mat}_2(\mathcal{O})$-order (which is known to be always principal). Find a right ideal generator α of I.

Problem 2.1.2 can be viewed as a generalization of the principal ideal problem over quaternion algebras over number fields. Let B be a quaternion algebra over a number field K, $\mathcal{O}$ be a maximal order in B, and I be a maximal ideal. An algorithm for finding a principal ideal generator can be dated back to Kirschmer and Voight in [KV10]. They gave algorithms both for totally definite quaternion algebras (when B is ramified at all infinite places) and indefinite quaternion algebras (when B is split at at least one infinite place). While the algorithm runs in deterministic polynomial time in the totally definite case, the authors were not able to provide a complexity analysis for the indefinite case. Page gave an algorithm in [Pag14] for the indefinite case, and under on various hypotheses on the distribution of units, class group elements, and powersmooth elements, he claimed that the algorithm runs in subexponential time. More recently, Hoffman and Johnston provided in [HJ20] a generic algorithm to tell whether two finitely generated modules over a semi-simple K-algebra are isomorphic, which includes solving principal ideal problems as a special case, however, their algorithm required some hypotheses on the algebra A, which includes a locally free cancellation property on the division algebras contained in A, and the property only holds for finitely many cases of quaternion algebras over $\mathbb{Q}$ (see [HM06]). Therefore, our situation is incompatible to the hypotheses in [HJ20] and the answer to Question 1.2 remains to be open.

2.1.3 Outline

In this paper, we will give an algorithm for the principal ideal generator problem over $\mathrm{Mat}_2(B_{p,\infty})$, the matrix ring of quaternion algebra over $\mathbb{Q}$, ramified at p and ∞.

Theorem 2.1.3. Suppose $B_{p,\infty}$ is a quaternion algebra over $\mathbb{Q}$, ramified exactly at p and ∞ for some prime p, $\mathcal{O}$ be a maximal order in $B_{p,\infty}$. Let I be a right $\mathrm{Mat}_2(\mathcal{O})$ order.

Then, under Heuristics 2.4.4, 2.5.11, and 2.5.6, Algorithm 2.3.1 gives a probabilistic algorithm which runs in subexponential time in p, and finds the principal ideal generator of I.

The approach will be similar to Page's. The algorithm will be split into a global part and a local part. In the global part, we will rescale the ideal I so that its norm is powersmooth, and then reduce the ideal by two-sided ideals so that we can shorten the computation in the local part and guarantee that the output is correct after the local reduction. For the local reduction, we need to reduce the ideal at each places dividing $N(I)$, the reduced norm of I. The upshot is that at each place ℓ, there is a equivalence between right $\mathbb{Z}_\ell$-ideals and Bruhat-Tits buildings of $\mathbb{Q}_\ell$. Therefore, when we act on the Bruhat-Tits building by a unit a maximal order in $\mathrm{Mat}_2(B_{p,\infty})$, we also have a corresponding action on the ideal side. Therefore, reducing a filtration in the Bruhat-Tits tree also helps reducing the ideals.

The remaining of the paper is laid out as follows. We give a brief overview of the necessary background in quaternion algebras, central simple algebras, and lattice theory over local fields in Section 2.2. The major steps for the principal ideal generator problem is stated in Algorithm 2.3.1 in Section 2.3. The first two major step of Algorithm 2.3.1, which involves reducing the input ideal I by two-sided ideals, is described in Section 2.4. After then we will need to reduce the simplified ideal I on each place dividing the norm of I, and this involves the local theory in Section 2.5. The validity of the algorithm and the complexity analysis is in Section 2.6, and Section 2.7 gives some experimental examples.

2.2 Background

2.2.1 General theory of central simple algebras

For more details on the topic, see [Rei03], which contains extensive theory on a central simple algebra A over F, where F is the fraction field of a Dedekind domain R. We will give a brief sketch of the general theory here, and then turn to two specific cases: the quaternion algebras in Section 2.2.2, and the matrix rings over quaternion algebras, in Section 2.2.3.

Let F be a field. We say that a finitely generated F-algebra A is a central simple algebra over F if $Z(A)$, the center of A, is equal to F, and A contains no non-trivial two-sided A-ideals. The degree, denoted as $[A : K]$, will be a square n^2. And the Artin-Wedderburn theorem implies that A is isomorphic to some matrix algebra $\mathrm{Mat}_m(D)$, where $n = md$, and D is a division ring over F such that $[D : F] = d^2$.

Suppose now that F is the fraction field of a Dekekind domain $\mathcal{O}_F$, then we say that I is an ideal in A if it is an $\mathcal{O}_F$-lattice satisfying $FI = A$. An order Λ is an ideal which is also a subring of A. And we call an order Λ maximal if it is not properly contained in another order of A. We call an ideal I a left (respectively, right) Λ-ideal, if $I\Lambda \subseteq I$ (respectively, $\Lambda I \subseteq I$). Given an ideal I, the left order of I is the order $O_l(I) := \{\alpha \in A \mid xI \subset I\}$ in A. We can similarly define a right order of I. We say that the ideal I is normal if both the left and right orders of I are maximal, and in this case, we call I a connecting ideal of $O_l(I)$ and $O_r(I)$. An we call a normal ideal I integral, if it is contained in its left order. For an ideal I, we can define its inverse ideal $I^{-1} = \{x \in A \mid IxI \subseteq I\}$.

We can define the reduced norms of an ideal I, which is the ideal generated by the reduced norms of elements in I.

For an element $\alpha \in A$, one can define a reduced norm $N(\alpha) \in F$ and a reduce trace $Tr(\alpha) \in F$ from the constant term and the second highest term in the reduced minimal polynomial (see [Rei03, (9.6)]). And for an ideal I in A, the reduced norm of the ideal I is defined as the ideal in F generated by reduced norms of elements in I. And we can also define a reduced discriminant from an ideal I. Suppose $[A : F] = n^2$. Then one can form an ideal $D(I)$ as the ideal in F generated by $\det(Tr(\alpha_i\alpha_j)_{i,j=1}^{n^2})$, where $\{\alpha_i\}_{i=1}^{n^2} \subset I$. From [Rei03, Corollary (25.10)], this $D(I) = d(I)^n$ for some ideal I, and we call $d(I)$ the reduced norm of the ideal I. It turns out that when we run through maximal orders Λ in A, the reduced norm $d(\Lambda)$ is an invariant. We call this invariant the reduced discriminant δ_A of A. And one defines the absolute discriminant of A as $\Delta_A d_F \delta_A^2$.

When the left and right order of $\mathfrak{I}$ coincide, say to Λ, then $\mathfrak{I}$ is said to be a two-sided Λ-ideal.

When A is a central simple algebra over a number field or a local field, we are in particular interested in the structure of the two-sided Λ-ideals. Let $\mathfrak{P} \subseteq \Lambda$ be a two-sided Λ-ideal. We call $\mathfrak{P}$ prime, if for any two-sided Λ-ideals $\mathfrak{I}, \mathfrak{J}$, if $\mathfrak{I}\mathfrak{J} \subseteq \mathfrak{P}$ implies either $\mathfrak{I} \subseteq \mathfrak{P}$ or $\mathfrak{J} \subseteq \mathfrak{P}$. [Rei03, Theorem (22.4)] gives a way to characterize the two-sided Λ-ideals, by constructing a bijection between the following:

$$\begin{array}{ccc} \{\text{Two-sided prime } \Lambda_p\}\text{-ideals} & \xrightarrow{\sim} & \{\text{Prime ideals in } F\} \\ \mathfrak{P} & \leftrightarrow & \mathfrak{p}, \end{array}$$

with the relation $\mathfrak{p} = \mathfrak{P} \cap \mathcal{O}_F$, and $\mathfrak{P} = \Lambda \cap \mathrm{rad}\Lambda_{\mathfrak{p}}$. Here rad is the Jacobson radical, defined as

$$\mathrm{rad}\Lambda = \bigcap_{L \text{ maximal left } \Lambda\text{-ideal}} \mathrm{ann}(\Lambda/L) = \{x \in \Lambda \mid 1 - axb \in \Lambda^\times \text{ for all } a, b \in \Lambda\}.$$

In addition, two-sided ideals in Λ have the “prime ideal factoring” theorem similar to the number field case: For two two-sided Λ-ideals $\mathfrak{I}, \mathfrak{J}$, $\mathfrak{I}\mathfrak{J} = \mathfrak{J}\mathfrak{I}$, and every ideal $\mathfrak{I}$ can be expressed as a product of two-sided prime ideals, and such an expression is unique up to permutation. In Page’s approach in [Pag14, SubAlgorithm 3.12] in the quaternion algebra case and our approach in the case of matrix ring over quaternion algebra, the factorization structure means that we can simplify a right Λ-ideal by extracting two-sided ideals, which turned out to have a better understood structure.

Proposition 2.2.1. Suppose again that A is a central simple algebra over F such that $[A : F] = n^2$, Λ is a maximal order of A, and $\mathcal{O}_F$ is a maximal order of F.

Let I be an integral right Λ-order in A. Suppose that as a module, I has a $\mathbb{Z}$-basis $\{v_1, \cdots, v_{n^2}\}$, and Λ has a $\mathbb{Z}$-basis $\{u_1, \cdots, u_{n^2}\}$. Then the smallest generating two-sided Λ-ideal is the ideal $\mathfrak{I}$ with $\{u_i v_j\}_{1 \leq i,j \leq n^2}$ as $\mathbb{Z}$-basis.

The above fact is then almost immediate from the definition. First, $\mathfrak{I}$ is indeed an ideal, and since $O_l(\mathfrak{I}) \supseteq \Lambda$ and $O_r(\mathfrak{I}) \supseteq \Lambda$, it is indeed a two-sided Λ-ideal. It is the smallest possible two-sided ideal containing I since all $u_i v_j$ must be such an ideal. Therefore, $I\mathcal{I}^{-1}$ is an integral right Λ-ideal and is not contained in any two-sided Λ-ideal.

Now we return to one-sided ideals, and construct the class set, which is the generalization of the class groups on a number field. Now, suppose A is a central simple algebra over a number field F, and $\mathcal{O}_F$ is a maximal order of F. We say that two maximal orders Λ, Λ' of A are conjugate, or of the same type, if there is an element $\alpha \in A^\times$ such that $\Lambda' = \alpha\Lambda\alpha^{-1}$. And suppose we fix a maximal order $\Lambda \subseteq A$. For two right (respectively, left) Λ-ideals I and I' of A, we say that I and I' are equivalent if there is an element $\alpha \in A^\times$ such that $I = I'\alpha$ (respectively, $I = \alpha I'$). This gives an equivalence relation, and we can therefore construct the ideal class set of right (respectively, left) Λ-ideals. We will be interested in the computational problem to determine whether two right (or left) Λ-ideals are equivalent. When one of the ideal is Λ, then the problem reduced to determining whether an ideal is principal.

It turns out that the Eichler's condition, which we will define below, plays a crucial role on the nature of the ideal class problem. In brief, the Eichler's condition indicates the cases when the strong approximation works. It is also the situation when we can relate one-sided ideal classes in the central simple algebra A with ideal classes in the underlying field F.

Definition 2.2.2 (Eichler's condition). Let A be a central simple algebra over a number field F. We say that A satisfy Eichler's condition over F if there is an infinite place v such that A_v^1 (the norm 1 elements of A) is not compact. Equivalently, A satisfied Eichler's condition if A is not a definite quaternion algebra over F.

The importance for A satisfying Eichler's condition is that the strong approximation applies:

Theorem 2.2.3 (Strong approximation and its consequences)**.** Let A be a central simple algebra satisfying Eichler's condition over F satisfying $(A : F) = n^2$. Let Λ and $\mathcal{O}_F$ be a maximal ideal of A and F, respectively. The following hold.

(i) (Strong approximation) The image of the diagonal embedding $\Lambda^1 \to \mathbb{A}_f^1$ is dense, where Λ^1 and $\mathbb{A}_{F,f}^1$ are the subgroup of norm 1 of Λ and the subgroup of the finite adele ring $\mathbb{A}_{F,f}$ of F of norm 1, respectively.

(ii) For a prime $\mathfrak{p}$ in F which splits in A, the map $\Lambda^1 \to \mathrm{SL}_n(\mathcal{O}_F/\mathfrak{p}^k)$ induced by completion and reduction is a surjection for any integer k.

(iii) The reduced norm map $\Lambda^\times \to \mathcal{O}_{F,A}^\times$ is surjective on the unit groups, where $\mathcal{O}_{F,A}^\times$ is the totally positive elements in $\mathcal{F}$ with respect to all infinite places in F splitting in A.

(iv) The reduced norm map induces a surjective map $\mathrm{Cl}(\Lambda) \to \mathrm{Cl}_A(\mathcal{O}_F)$ from the class set to Λ to the ray class group of F with the conductor being the product of all infinite places in F splitting in A.

Proof. See [Rei03, theorem 34.9]. □

A quick consequence of Theorem 2.2.7 is that if A is a central simple algebra over $\mathbb{Q}$ which satisfies the Eichler's condition, with a given maximal ideal Λ, then every one-sided Λ-ideal is principal.

2.2.2 Quaternion algebras and supersingular elliptic curves

A standard reference of the topic is [Voi20]. Let F be a generic field of characteristic not equal to 2. A quaternion algebra B over F is a central simple algebra satisfying $[B : F] = 4$. Equivalently, B is an algebra which can be expressed in the form $B = \{F + Fi + Fj + Fk \mid i^2 = a, j^2 = b, ij = -ji = k\}$ for some $a, b \in F^\times$. A quaternion algebra is either isomorphic to the matrix algebra $\mathrm{Mat}_2(F)$ or is a division ring. For any element $\alpha := w + xi + yj + zk$, there is a unique involution $\bar{\alpha} := w - xi - yj - zk$, and we can define a reduced norm $N(\alpha) = \alpha\bar{\alpha}$ and reduced trace $Tr(\alpha) = \alpha + \bar{\alpha}$.

In this paper, we are interested in quaternion algebras over number fields and their completions. Now, suppose B is a quaternion algebra over the number field F with ring of integers $\mathcal{O}_F$. At each place v of F, we say that B is ramified at v if the completion $B_v := B \otimes_F F_v$ is a division algebra; otherwise we say that B splits at v. And we say that the quaternion algebra B is totally definite if B ramifies at all archimedean places; otherwise we say that B is indefinite. For a quaternion algebra B, the reduced discriminant, as described in Section 2.1, is the product of all the primes in F which are ramified in B.

As quaternion algebras are central simple algebras, we refer to Section 2.1 for the definitions of orders and ideals, type sets of B and class sets of maximal orders $\mathcal{O}$ of B.

Next, we look at another object, supersingular elliptic curves over the closure of finite field $\overline{\mathbb{F}_p}$, and discuss the connections with the quaternion algebra $B_{p,\infty}$, the quaternion algebra ramified exactly at the place p and the archimedean place. A general discussion of the topic can be found in [Wat69].

Suppose E is an ellitic curve, defined over the algebraic closure of a finite field $\overline{\mathbb{F}_p}$. E is defined to be ordinary if the p-torsion points $|E[p]| = p$; and is defined to be supersingular if $|E[p]| = 1$. Equivalently, E is ordinary when the full endomorphism algebra $\mathrm{End}^0_{\overline{\mathbb{F}_p}}(E) := \mathrm{End}_{\overline{\mathbb{F}_p}}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$ is a quadratic field; and is supersingular when $\mathrm{End}^0_{\overline{\mathbb{F}_p}}(E)$ is isomorphic to $B_{p,\infty}$.

2.2.3 Central simple algebras and superspecial abelian varieties

In the case of dimension 2, we say that an abelian vaiety over the field $\overline{\mathbb{F}_p}$ is supersingular, if it is $\overline{\mathbb{F}_p}$-isogenous to a product of supersingular elliptic curves; and an abelian surface is superspecial, if it is isomorphic to a product of supersingular elliptic curves.

It can be shown that there is only one $\overline{\mathbb{F}_p}$-isomorphism class of superspecial abelian variety. Indeed, it is a result by Deligne, Ogus and Shioda that when $g \geq 2$, if $E_1, \cdots, E_{2g}$ are supersingular elliptic curves over $\overline{\mathbb{F}_p}$, then $E_1 \times \cdots \times E_g \cong E_{g+1} \times \cdots \times E_{2g}$ (see [Shi79, Theorem 3.5]). Therefore, we can choose a supersingular elliptic curve E over $\mathbb{F}_p$, and every supersingular (respectively, superspecial) abelian surface is isogenous (respectively, isomorphic) to E^2 over the algebraic closure. Consequently, later we will see that to make sense of the isomorphism classes of superspecial abelian varieties, we need to consider the principal polarizations as well.

We are also interested in the endomorphism ring of a supersingular or superspecial abelian variety. It is known from [Wat69] that an abelian variety of dimension g is supersingular if and only if $\dim_{\mathbb{Q}} \mathrm{End}^0_{\overline{\mathbb{F}_p}}(E) = (2g)^2$. The problem is to find their endomorphism rings.

Let E be a supersingular elliptic curve as above, and denote $\mathcal{O} = \mathrm{End}_{\overline{\mathbb{F}_p}}(E)$ and $B_{p,\infty} = \mathcal{O} \otimes_{\mathbb{Z}} \mathbb{Q}$. Then the endomorphism ring of a superspecial abelian variety is as below:

Proposition 2.2.4. Suppose $A = E^2$ is a superspcial abelian surface. Then $\mathrm{End}_{\overline{\mathbb{F}_p}}(A) = \mathrm{Mat}_2(\mathcal{O})$, and it is embedded in $\mathrm{End}^0_{\overline{\mathbb{F}_p}}(A_0) = \mathrm{Mat}_2(B_{p,\infty})$.

Proof. Indeed, this can be verified by observing that every element $\Psi := \begin{bmatrix} \psi_{11} & \psi_{12} \\ \psi_{21} & \psi_{22} \end{bmatrix} \in$ $\mathrm{Mat}_2(\mathcal{O})$ gives an endomorphism $\Psi : A \to A$, $(P, Q) \mapsto (\psi_{11}(P) + \psi_{12}(Q), \psi_{21}(P) + \psi_{22}(Q))$, and $\mathrm{End}_{\overline{\mathbb{F}_p}}(A)$ cannot be bigger than that since $\mathrm{Mat}_2(\mathcal{O})$ is already a maximal order in $\mathrm{Mat}_2(B_{p,\infty})$ by [Rei03, Theorem (8.7)]. $\square$

When the supersingular E and its endomorphism ring $\mathcal{O}$ is fixed, then we can also fix the endomorphism algebra $\mathrm{Mat}_2(B_{p,\infty})$. Then, when we have another superspecial abelian surface A', constructed from $A = E^2$ via an isomorphism $E^2 \to A'$, then the endomorphism ring of A' can be characterized as below.

Proposition 2.2.5. Let $E^2 \to A'$ be a separable isogeny for some superspecial abelian variety A'. Then $\mathrm{End}_{\overline{\mathbb{F}_p}}(A')$ is a maximal order of $\mathrm{End}_{\overline{\mathbb{F}_o}}(A') \otimes_{\mathbb{Z}} \mathbb{Q} \cong \mathrm{Mat}_2(B_{p,\infty})$, $I = \mathrm{Hom}(A', E^2)$ is a linking order of $\mathrm{End}_{\overline{\mathbb{F}_p}}(E^2) = \mathrm{Mat}_2(\mathcal{O})$ and $\mathrm{End}_{\overline{\mathbb{F}_p}}(A')$. There exists a generating bijection $h \in I$, so that $I = \mathrm{End}_{\overline{\mathbb{F}_p}}(E^2)h = h\mathrm{End}_{\overline{\mathbb{F}_p}}(A')$, and $\mathrm{End}_{\overline{\mathbb{F}_p}}(A') = h^{-1}\mathrm{End}_{\overline{\mathbb{F}_p}}(E^2)h$.

In particular, if $E^2 \to A'$ is an automorphism, then we have a embedding of $\mathrm{End}_{\overline{\mathbb{F}_p}}(A') \otimes_{\mathbb{Z}} \mathbb{Q}$ in $\mathrm{Mat}_2(B_{p,\infty})$ as a maximal order.

In contrast, if A is a supersingular surface which is not superspecial, the endomorphism ring will be smaller in general: the p-part of $\mathrm{End}_{\overline{\mathbb{F}_p}}(A)$ is only a subgroup of the p-part of a conjugate of $\mathrm{Mat}_2(\mathcal{O})$. Yu-Yu computed the endomorphism ring of supersingular abelian surfaces in [YY09, Proposition 3.2].

Indeed, this endomorphism rings of superspecial abelian varieties behaved very differently from the genus 1 case, since we know that as $\mathrm{Mat}_2(B)$ satisfies the Eichler's condition, it has class number 1, see the discussion at the end of Proposition 2.2.1.

Since there is only one isomorphism of superspecial abelian variety over $\overline{\mathbb{F}_p}$, which is E^2, we actually need to consider the principal polarizations to form meaningful isomorphism classes. Indeed, in [IKO86, section 2], it is shown that polarizations can be transferred as matrices in $\mathrm{End}_{\overline{\mathbb{F}_p}}(E^2)$. And we will establish a few equivalent maps between isomporphism classes of principally polarized superspecial abelian surfaces and isomorphism classes in certain central simple algebra.

We first fix a supersingular elliptic curve E over $\mathbb{F}_p$ as before. For the abelian variety $A = E^2$, we denote a polarization of A by an ample divisor L. And the divisor naturally defines an isogeny $\varphi_L : A \to A^\vee$ by $x \mapsto T_x^* L - L$, where $A^\vee$ is the dual abelian variety of A, and T_x is the "translation by x" map. The degree of the divisor L is known to be the degree of the isogeny φ_L.

In particular, when $A = E^2$, there is a product polarization given by the divisor $L_0 = E \times \{0\} + \{0\} \times E$. L_0 is then a a divisor of degree 1. Then we define the following map:

Definition 2.2.6. Denote $\mathrm{End}^s(A)$ to be the endomorphisms of A fixed by the Rosati involution (defined with respect to the polarization L_0). Denote by $\mathrm{Pic}(A)$ and $\mathrm{Pic}^0(A)$ the Picard group of A and the connected component of $\mathrm{Pic}(A)$ at the origin. Then $\mathrm{Pic}(A)/\mathrm{Pic}^0(A)$ is the Néron-Severi group of A, which we denote by $\mathrm{NS}(A)$. And we have a map

$$\begin{aligned} j : \mathrm{NS}(A) &\xrightarrow{\sim} \mathrm{End}^s(A) \\ L &\mapsto \varphi_{L_0}^{-1} \circ \varphi_L, \end{aligned}$$

Theorem 2.2.7 ([IKO86, Corollary 2.9]). The isomorphism classes of principal polarizations are bijective to

$$\left\{ \begin{bmatrix} s & r \\ \bar{r} & t \end{bmatrix} \in \mathrm{Mat}_2(\mathcal{O}) \;\middle|\; s, t \in \mathbb{Z}_{>0}, st - r\bar{r} = 1 \right\} \Big/ \sim,$$

where we call $f_1 \sim f_2$ if and only if there is an $\alpha \in \mathrm{End}(A)$ satisfying $f_1 = \alpha^\dagger f_2 \alpha$, where $\alpha^\dagger = \overline{\alpha}^t$ corresponds to the Rosati involution of α with respect to the polarization L_0.

Proof. Given principal polarized abelian varieties (A, L) and (A', L'), we say they are isomorphic if and only there exists an isomorphism $\alpha : A \to A'$ such that the diagram below commutes:

$$\begin{tikzcd} A \arrow[r, "\alpha"] \arrow[d, "\varphi_L"'] & A' \arrow[d, "\varphi_{L'}"] \\ A^\vee & (A')^\vee \arrow[l, "\alpha^\vee"] \end{tikzcd}$$

Back to our scenario, when the polarizations L_1 and L_2 of E^2 are equivalent, there exists an $\alpha \in \mathrm{End}_{\overline{\mathbb{F}_p}}(E^2)^\times$ satisfying $\varphi_{L_2} = \alpha^\vee \circ \varphi_{L_1} \circ \alpha$. As φ_{L_0} is an isomorphism between E^2 and $(E^2)^\vee$, we have $\varphi_{L_0}^{-1}\varphi_{L_2} = (\varphi_{L_0}^{-1}\alpha^\vee\varphi_{L_0}) \circ (\varphi_{L_0}^{-1}\varphi_{L_1}) \circ \alpha$, or $j(L_2) = \alpha^\dagger \circ j(L_1) \circ \alpha$.

The theorem then follows by fixing a supersingular elliptic curve E over $\overline{\mathbb{F}_p}$, taking $\mathcal{O} = \mathrm{End}_{\overline{\mathbb{F}_p}}(E)$ and $B = \mathcal{O} \otimes_{\mathbb{Z}} \mathbb{Q}$, and identifying $\mathrm{End}_{\overline{\mathbb{F}_p}}(E^2)$ with $\mathrm{Mat}_2(\mathcal{O})$. □

Definition 2.2.8. We denote by $\mathrm{Mat}_2^1(\mathcal{O})$ the matrices in $\mathrm{Mat}_2(\mathcal{O})$ with reduced norm ± 1, and by $\mathrm{Mat}_2^+(\mathcal{O})$ the matrices satisfying the conditions in Theorem 2.2.7. For $g_1, g_2 \in \mathrm{Mat}_2^+(\mathcal{O})$, $g_1 \sim g_2$ if there exists $\gamma \in \mathrm{Mat}_2(\mathcal{O})^\times$ satisfying $\overline{\gamma}^t g_1 \gamma = g_2$.

So far all the objects involved in the arithmetic geometry side are abelian surfaces with a principal parametrization. For the practical purpose, we need to translate them to objects which can actually be computed. In dimension 2, we know that every principally polarized superspecial abelian surface $(A = E^2, L)$ is isomorphic either to $\mathrm{Jac}(C)$ for some genus 2 hyperelliptic curve C or to $E_1 \times E_2$ for some elliptic curves E_1, E_2 ([FT19, Theorem 1]). Therefore, combining Definitions 2.2.6, 2.2.8 and Theorem 2.2.7, we have the following bijections between sets

$$\begin{aligned} &\left\{\begin{matrix} \mathrm{Jac}(C), C\text{: superspecial} \\ \text{hyperellipic, genus 2 or} \\ E_1 \times E_2, E_1, E_2\text{: supersingular elliptic} \end{matrix}\right\}_{/\sim} \leftrightarrow \left\{\begin{matrix} \text{principal} \\ \text{polarizations} \\ \text{of } E^2 \end{matrix}\right\}_{/\sim} \\ &\qquad\qquad\qquad\qquad\qquad\qquad \leftrightarrow \mathrm{NS}^1(E^2) \quad \leftrightarrow \mathrm{Mat}_2^+(\mathcal{O})_{/\sim}. \end{aligned} \tag{2.1}$$

In particular, Equation (2.1) gives a correspondence between isomorphism between superspecial Jacobians or product of supersingular elliptic curves (the abelian variety side) to objects in the central simple algebra $\mathrm{Mat}_2^+(\mathcal{O})$ (the "endomorphism ring" side). And we claim that this indeed provides a better generalization to the supersingular elliptic curve case then supersingular abelian varieties. One of the reasons is, the dimension of the supersingular locus of abelian varieties of dimension g is $\lfloor g^2/4 \rfloor$ in the coarse moduli space of abelian varieties of dimension g (see [LO98, Corollary 4.4]). Therefore, when $g \geq 2$, there are infinitely many supersingular abelian varieties over $\overline{\mathbb{F}_p}$. On the other hand, on the superspecial abelian variety side, we know that all superspecial abelian varieties are defined over $\mathbb{F}_{p^2}$, and there are formulas which computes both the number of equivalence classes of polarization over $\mathbb{F}_{p^2}$ and the number of equivalence classes which can be defined over $\mathbb{F}_p$, see [Ibu19]. In addition, we know that every polarized abelian variety $(A = E^2, L)$ is isomorphic either to $\mathrm{Jac}(C)$ for some genus 2 hyperelliptic curve C or to $E_1 \times E_2$ for some elliptic curves E_1, E_2 ([FT19, Theorem 1]). This enables us to compute the objects explictly.

Now, the ultimate goal is to make an analogy between the genus 1 theory and genus 2 theory by "replace the maximal orders and ideals in genus 1 by matrices in $\mathrm{Mat}_2^+(\mathcal{O})$ in genus 2". For this, we need to generalize the ℓ-isogeny graphs in supersingular abelian varieties. In the elliptic curve case, we have the explicit Vélu formula, which computes an ℓ-isogeny in time $O(\ell)$. However, in the abelian variety case, not all isogenies are easy to compute–in some cases taking a general isogeny even makes the destination abelian surface no longer principally polarizable. Therefore, we shall restrict to the special case: the (ℓ, ℓ)-isogenies. The following Proposition demonstrates that we can also translate an (ℓ, ℓ)-isogeny of principally polarized abelian surfaces to maps in equivalence classes of $\mathrm{Mat}^+(\mathcal{O})$.

Proposition 2.2.9 (Relation between (ℓ, ℓ)-isogenies and $\mathrm{Mat}_2^+(\mathcal{O})$)**.** Let the setting be the same as above. Suppose $\gamma : (E^2, L_1) \to (E^2, L_2)$ is an (ℓ, ℓ)-isogeny, and $g_1, g_2 \in \mathrm{Mat}_2^+(\mathcal{O})$ are representatives of (E^2, L_1) and (E^2, L_2) in the bijection described in Theorem 2.2.7, respectively. Then by identifying $\mathrm{End}_{\overline{\mathbb{F}_p}}(E^2)$ and $\mathrm{Mat}_2(\mathcal{O})$, $\gamma \in \mathrm{Mat}_2(\mathcal{O})$ has reduced norm ℓ^2 and satisfies

$$\overline{\gamma}^t g_2 \gamma = \ell g_1.$$

Proof. If the polarizations (E^2, L_1) and (E^2, L_2) are compatible with respect to the (ℓ, ℓ)-isogeny γ, the following diagram is commutative:

$$\begin{tikzcd}
E^2 \arrow[d, "\varphi_{L_1}"'] & E^2 \arrow[l, "{[\ell]}"'] \arrow[r, "\gamma"] & E^2 \arrow[d, "\varphi_{L_2}"] \\
(E^2)^\vee & & (E^2)^\vee \arrow[ll, "\gamma^\vee"]
\end{tikzcd}$$

Therefore, $\varphi_{L_1} \circ [\ell] = \gamma^\vee \circ \varphi_{L_2} \circ \gamma$, or $(\varphi_{L_0}^{-1} \varphi_{L_1}) \circ [\ell] = (\varphi_{L_0}^{-1} \gamma^\vee \varphi_{L_0}) \circ (\varphi_{L_0}^{-1} \varphi_{L_2}) \circ \gamma$, which yields to the conclusion. □

At this stage, although there are still some obstructions, we have a dictionary book which maps objects related to supersingular elliptic curves to superspecial abelian varieties, and ℓ-isogenies on elliptic curves corresponds to (ℓ, ℓ)-isogenies on abelian surfaces. By considering the endomorphism ring, maximal orders in B becomes equivalence classes of $g \in \mathrm{Mat}_2^+(\mathcal{O})$; and connecting ideals I in B becomes $\gamma \in \mathrm{Mat}_2(\mathcal{O})$.

We proposed to generalize the signature algorithm by Galbraith et al. to genus 2. See Appendix 1 for a sketch and [GPS19] for more details. For the central zero-knowledge identification scheme in genus 2, when we replace the "Isogeny to Ideal" algorithm in [GPS19, Section 4.4], we needed the algorithm `IsogenyToMatrix`, which translates an isogeny path to a matrix in $\mathrm{Mat}_2^+(\mathcal{O})$. The proposed genus 2 algorithm is in Algorithm 2.1. We see that step 17 in the algorithm can be solved if we can find a principal ideal generator of $\Gamma\mathrm{Mat}_2(\mathcal{O}) + p_j\mathrm{Mat}_2(\mathcal{O})$, hence leads to the principal ideal problem (Problem 2.1.2).

2.2.4 Lattices over a local field and Bruhat-Tits buildings

Let K be a non-archimedean local field whose residue field has q elements, let $\mathcal{O}_K$ be its ring of integers, and let π be a uniformizer of K. We first define the Bruhat-Tits building on $\mathrm{PGL}_4(K)$. Much of the theory can be found in [KL14].

Definition 2.2.10 (The Bruhat-Tits building on $\mathrm{PGL}_4(K)$)**.** The Bruhat-Tits building is a directed graph $\mathcal{T}_K$ with the following structure:

(i) We say that two $\mathcal{O}_K$-lattices P' and P'' are homothetic, or in the same homothety class, if there is an $\alpha \in K^\times$ such that $P' = \alpha P''$. Vertices of $\mathcal{T}_K$ are the homothety classes of $\mathcal{O}_K$-lattices of rank 4 which is embedded in K^4, a fixed K-vector space of dimension 4.

(ii) We define a $\mathcal{O}_K$-lattice P_0 which has $\{e_{K,i}\}_{i=1}^4$ as a basis, where $e_{K,i} \in K^4$ forms the standard basis. The homothety class $[P_0]$ is considered as the central element of the Bruhat-Tits building $\mathcal{T}_K$.

(iii) For two homothety class $[P'] \neq [P'']$, there is an edge $[P'] \to [P'']$, if there are lattices P' and P'' in each homothety class satisfying $\pi P' \subset P'' \subset P'$. Suppose that $\pi P' \subset P'' \subset P'$. We further classify an edge $[P'] \to [P'']$ to be type 1, 2, or 3 according to whether $[P' : P'']$ is equal to q, q^2, or q^3.

(iv) For an ordered quadruple of vertices in $\mathcal{T}_K$, $([P], [P'], [P''], [P'''])$, we say that it forms a chamber if all the four ordered pairs $[P] \to [P'], \cdots, [P'''] \to [P]$ are connected with type 1 edges.

From the definition above, we know that on any vertex, there are $q^3 + q^2 + q + 1$, $(q^2 + 1)(q^2 + q + 1)$, and $q^3 + q^2 + q + 1$ out vertices of type 1, 2, 3 from that vertex, respectively.

We can define a left $\mathrm{GL}_4(K)$-action on rank 4 $\mathcal{O}_K$-lattices as well as vertices in $\mathcal{T}_K$ as follows. Let $g = (g_{ij})_{1 \leq i,j \leq 4} \in \mathrm{GL}_4(K)$. Suppose $e_1, \cdots e_4$ is a basis of a lattice P. Then $g \cdot P$ is the lattice generated by $\sum_{j=1}^4 g_{ij} e_j$, $i = 1, \cdots, 4$. Under this setting, by rewriting the ideas in [KL14] to the scenario of $\mathrm{GL}_4(K)$, we can characterize the stabilizer of objects in the Bruhat-Tits building as follows:

Proposition 2.2.11. Let $\mathcal{T}_K$ and P_0 be as in Definition 2.2.10.

(i) The action of $\mathrm{Mat}_4(K)$ is transitive of the vertices, edges and chambers in $\mathcal{T}_K$.

(ii) Suppose P is a vertex in $\mathcal{T}_K$, with $g \cdot P_0 = P$ for some $g \in GL_4(K)$. Then the stabilizer of $[P]$ is $g(K^\times \mathrm{GL}_4(\mathcal{O}_K))g^{-1}$.

(iii) Let $\sigma = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \pi & 0 & 0 & 0 \end{bmatrix}$. Then $[P_0] \to [\sigma \cdot P_0]$ forms a type 1 edge. If $[Q_0] \to [Q_1] = g \cdot ([P_0] \to [\sigma \cdot P_0])$ is an arbitrary type 1 edge, then the stabilizer of the edge $[Q_0] \to [Q_1]$

is $g(K^\times \mathscr{E})g^{-1}$, where

$$\mathscr{E} = \left\{ g \in \mathrm{GL}_4(\mathcal{O}_K) \,\middle|\, g \equiv \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ 0 & 0 & 0 & * \end{bmatrix} \pmod{\pi \mathrm{GL}_4(\mathcal{O}_K)} \right\}.$$

(iv) $([P_0], [\sigma \cdot P_0], \cdots, [\sigma^3 \cdot P_0])$ forms a chamber.

For an arbitrary chamber $([Q_0], [Q_1], \cdots, [Q_3]) = g \cdot ([P_0], [\sigma \cdot P_0], \cdots, [\sigma^3 \cdot P_0])$, its stabilizer is given by is $g(K^\times \mathscr{B})g^{-1}$, where

$$\mathscr{B} = \left\{ g \in \mathrm{GL}_4(\mathcal{O}_K) \,\middle|\, g \equiv \begin{bmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \\ 0 & 0 & 0 & * \end{bmatrix} \pmod{\pi \mathrm{GL}_4(\mathcal{O}_K)} \right\}.$$

Proof. (iii) and (iv) are true since $\mathscr{E} = \mathrm{GL}_4(\mathcal{O}_K) \cap \sigma \mathrm{GL}_4(\mathcal{O}_K)\sigma^{-1}$, and

$$\mathscr{B} = \mathrm{GL}_4(\mathcal{O}_K) \cap \sigma \mathrm{GL}_4(\mathcal{O}_K)\sigma^{-1} \cap \cdots \cap \sigma^3 \mathrm{GL}_4(\mathcal{O}_K)\sigma^{-3}.$$

□

So far we know that there is an action of local objects $K^\times \mathrm{GL}_4(\mathcal{O}_K)$ on a rank 4 $\mathcal{O}_K$-lattice. Now we also turn back to our global scenario. Again, denote $B = B_{p,\infty}$ be a quaternion algebra over $\mathbb{Q}$, ramified exactly at p and ∞, $\mathcal{O}$ be a maximal order of B. Then we take $A = \mathrm{Mat}_2(B)$, and $\Lambda = \mathrm{Mat}_2(\mathcal{O})$ be a maximal order of A. Also, denote Λ^1 to be the subgroup of Λ with reduced norm 1.

When $\ell \neq p$ is a prime splitting in the quaternion algebra B, we know that $\mathcal{O} \otimes_{\mathbb{Z}} \mathbb{Z}_\ell \cong \mathrm{Mat}_2(\mathbb{Z}_\ell)$ and $\Lambda \otimes_{\mathbb{Z}} \mathbb{Z}_\ell \cong \mathrm{Mat}_4(\mathbb{Z}_\ell)$. Fix an embedding $\iota : \Lambda \to \mathrm{Mat}_4(\mathbb{Z}_\ell)$. Then for $c \in \Lambda$ and P a rank 4 $\mathbb{Z}_\ell$-lattice, we can define $c \cdot P = \iota(c) \cdot P$. An important fact is that this action induces a transitive action on the type 1 neighbors of $[P_0]$ in the Bruhat-Tits building $\mathcal{T}_{\mathbb{Z}_\ell}$ (recall that $[P_0]$ is the assigned "center" of $\mathcal{T}_{\mathbb{Z}_\ell}$).

Proposition 2.2.12. Λ^1, the unit norm group of Λ, acts transitively on the type 1 neighbors of P_0 in the Bruhat-Tits building $\mathcal{T}_{\mathbb{Z}_\ell}$.

Proof. Since $\iota(\Lambda^1) \subseteq \mathrm{GL}_4(\mathbb{Z}_\ell)$, Λ^1 stabilizes the center vertex P_0. Let P_1 and P_2 be lattices such that $[P_0 : P_1] = [P_0 : P_2] = \ell$. We want to find $c \in \Lambda^1$ such that $c \cdot P_1 = P_2$.

Let $\{e_1, \cdots, e_4\}$ be a basis of P_0. It is known that there exists bases $\{e'_1, \cdots, e'_4\}$ and $\{e''_1, \cdots, e''_4\}$ of P_0 such that $\{e'_1, \cdots e'_3, \ell e'_4\}$ is a basis of P_1 and $\{e''_1, \cdots e''_3, \ell e''_4\}$ is a basis of P_2. Suppose $g' = (g'_{ij}), g'' = (g''_{ij}) \in \mathrm{Mat}_4(\mathbb{Z}_\ell)$ satisfy $e'_i = \sum_{j=1}^4 g'_{ij} e_j$, then $P_2 = g'' g'^{-1} \cdot P_1$. Then $g'' g'^{-1} \in \mathrm{GL}_4(\mathbb{Z}_\ell)$, and up to a scaling of bases we can further make $g'' g'^{-1} \in \mathrm{SL}_4(\mathbb{Z}_\ell)$.

As the central simple algebra A satisfies the Eichler's condition, we know by Theorem 3(ii) that the action of Λ^1 on the neighbors of $[P_0]$ is transitive. □

2.3 The Principal Ideal Generator Algorithm

In this section, we describe the major steps of the principal ideal generator algorithm. After the algorithm, we give comments on each step, together with a pointer on where the steps will be explained. The validity and complexity of the main algorithm will be proven in Theorem 2.6.1 in Section 2.6.

Algorithm 2.3.1 (The main algorithm)**.**
Input: The quaternion algebra $B_{p,\infty}$ over $\mathbb{Q}$ which ramifies at p and ∞, a maximal order $\mathcal{O}$ of $B_{p,\infty}$, $A = \mathrm{Mat}_2(B_{p,\infty})$, a central simple algebra over $\mathbb{Q}$, and $\Lambda = \mathrm{Mat}_2(\mathcal{O})$, a maximal order in A. I, a right-Λ ideal.
Output: An ideal generator $\alpha \in I$, such that $I = \alpha\Lambda$.

(1) Search randomly for an element $s \in I^{-1}$ such that the reduced norm of sI is $\mathcal{B}$-smooth (i.e. all prime divisors of the reduced norm are in $\mathcal{B}$).

(2) (The `GReduce` routine.) Find a decomposition of the form $fcsI = J\mathfrak{J}$, where J is an integral right Λ-ideal, $\mathfrak{J}$ is a two-sided fractional Λ-ideal, and J is not contained in any non-trivial two-sided integral Λ-ideal. And $f \in \mathbb{Q}$, $c \in \Lambda$, which makes the reduced norm of both sides equal to 1.

(3) (The `LReduce` routine) Let $\mathcal{D}$ be the set of prime divisors of the reduced norm of J. For each prime $\ell \in \mathcal{D}$, perform a "local reduction at ℓ" as follows. Find a $c_\ell \in \Lambda^\times$ such that $c_\ell J = \ell^r J'$, where r is a non-negative integer an J' is a right integral Λ-ideal whose ℓ-valuation of the reduced norm is the smallest possible. Then, replace c by c_ℓ, replace J by J', and replace $\mathfrak{J}$ by $\ell^r \mathfrak{J}$.

(4) After handling all the possible local reductions, as all right Λ-ideals are principal, we should have $J = \mathfrak{J} = \Lambda$. The elements on the left-hand side provide a principal ideal generator for the ideal I.

Below are some brief comments of each substep and a pointer to the relevant section.

(1) We test elements in I^{-1} with small entries. Increasing the size of $\mathcal{B}$ speeds up this step but slows down the following steps. This step, together with the pre-processing part of the algorithm and initiations of parameters including the set $\mathcal{B}$, will be discussed in Section 2.4.1.

(2) This step is essentially identical to the GReduce algorithm in [Pag14]. Our `GReduce` routine is a close modification of [Pag14] and will be described in Section 2.4.2.

(3) We will need the compatibility between Λ-action on the ideals and on the Bruhat-Tits buildings. This will be discussed in Section 2.5.

(4) This step is comparatively straightforward. We will discuss this in Section 2.4, but most of them follow from the same reason as in [Pag14, proposition 3.13].

2.4 The Global Reductions of Ideals

2.4.1 The G-reduction structure

As in the previous sections, we consider the more special case, where $B = B_{p,\infty}$ is the quaternion algebra over $\mathbb{Q}$, ramified exactly at p and ∞, and $\mathcal{O}$ is a fixed maximal order of B. Taking $A = \mathrm{Mat}_2(B)$, we aim to establish a principal ideal generator algorithm on $\Lambda = \mathrm{Mat}_2(\mathcal{O})$. Also, we let Δ be the discriminant of B. Under this setting, by the consequences of strong approximation (Theorem 2.2.3), since $\mathbb{Z}$ has class number 1, so does Λ, and all the right Λ-ideals are principal.

Also, similar to Buchmann's class number algorithm in the case of number fields in [Buc88] and Page's principal ideal algorithm in the case of quaternion algebras in [Pag14], we need to define a set of small primes $\mathcal{B}$. We will elaborate practical considerations of choosing $\mathcal{B}$ in Section 2.4.

To find the left ideal generator for a given right Λ-ideal I, it would be preferable to reduce the ideal I to accelerate the local reduction process in Section 2.5.

More precisely, this contains two processes. First, we want the integral ideal I to be "smooth", which means that the prime divisors of $N(I)$ need to be in a prescribed prime set, so that we can pre-compute the necessary data for local reduction on these primes. We make an ideal smooth by replacing I with $I' = sI$ for a suitable randomly selected s. More details will be provided at the end of Section 2.4.1.

Second, starting from a smooth integral ideal I, since we saw in Section 2 that the two-sided ideals in $\mathrm{Mat}_2(B)$ are easier to handle, we want to rescale and decompose I as the form $fcI = J\mathfrak{J}$, where $\mathfrak{J}$ and $f \in \mathbb{Q}$ are two-sided; and J and c are one-sided. The hope is that the remaining unprocessed one-sided part J has a norm as simple as possible to reduce the workload in the two-sided reduction part. To finish this, we need a pre-computed G-reduction structure, which we will define below and give algorithms in Sections 2.4.1 and 2.4.2 to compute this.

We need the following G-reduction structure altered from [Pag14] for the reduction.

Definition 2.4.1 (G-reduction structure in A)**.** In the $A = \mathrm{Mat}_2(B_{p,\infty})$ setting, the G-reduction structure is obtained by computing the following data:

(i) Define a set $\mathcal{B}$ which contains primes ℓ up to a number M, excluding the ramified prime p.

(ii) Define a set $X \subset \Lambda^\times$ as follows. For each prime $\ell \in \mathcal{B}$, find a element $c_\ell \in \Lambda$ such that $N(c_\ell) = \ell$.

For the data in Definition 2.4.1, the relation between the choice of $\mathcal{B}$ in part (i) and the complexity of the algorithms will be discussed in Proposition 2.4.5. To generate the set X in part (ii), the general idea is to progressively pick elements of small entries until we find enough of them whose combination satisfies the criteria for X.

Before describing an algorithm that constructs X, we need to specify how to pick elements of small entries. One can define a positive definite quadratic form $Q : A \to \mathbb{R}$ on the central simple algebra A as $Q(m) = \sum_{1 \le i,j \le 2} \mathrm{Nred}(m_{ij})$, where m_{ij} is the ij-th entry of the 2×2 matrix m.

Using lattice reduction algorithms, one can enumerate elements in any $\mathbb{Z}$-lattice $M \subset \mathrm{Mat}_2(B)$ by sorting $Q(m)$ for $m \in M$ in increasing order. From the enumeration, we have a function `NextElement`(M) which returns the element in M which comes next in the enumeration (assuming that we keep a pointer in M storing the last element enumerated).

Now, given the function `NextElement`, we can describe the following algorithm, which generates the set X which has the property as in Definition 2.4.1.

Algorithm 2.4.2 (Constructing the set X in the G-reduction structure)**.**
Input: $\mathcal{B}$, a set of small prime.
Output: A set $X \subset A^\times$, which contains an element $c_\ell \in A$ satisfying $N(c_\ell) = \ell$ for each prime $\ell \in \mathcal{B}$.

```
Let $E \leftarrow \{\mathrm{diag}(\ell, \ell) \mid \ell \in \mathcal{B}\}$, $X \leftarrow \emptyset$.
while $\langle N(E) \rangle \subsetneq \langle \mathcal{B} \rangle$ do
    $\ell \leftarrow$ the smallest prime divisor in $\langle \mathcal{B} \rangle \backslash \langle N(E) \rangle$.
    $x \leftarrow$ LDivisibleElement$(\ell)$.
    if $N(x) \notin \langle N(E) \rangle$ then
```

```
 6:         $E \leftarrow E \cup \{x\}$
 7:     end if
 8: end while
 9: for $\ell \in \mathcal{B}$ do
10:     $c_\ell \leftarrow$ an element in $\langle E\rangle$ with reduced norm $\ell$.
11:     $X \leftarrow X \cup \{c_\ell\}$
12: end for
13: return $X$.
```

Algorithm 2.4.3 (The routine `LDivisibleElement` used in Algorithm 2.4.2).
Input: A prime ℓ.
Output: An element $x \in A$ with a small $\mathcal{B}$-smooth norm divisible by ℓ.

```
$\mathfrak{P} \leftarrow$ a right $\mathcal{O}$-ideal of norm $\ell$, $I \leftarrow \begin{bmatrix}\mathcal{O} & \mathcal{O}\\ \mathfrak{P} & \mathfrak{P}\end{bmatrix}$.
repeat
    $x \leftarrow$ NextElement$(I)$
until $x$ is $\mathcal{B}$-smooth
return $x$.
```

Heuristic 2.4.4. Define $L(x) := \exp(\sqrt{\ln x \ln\ln x})$. Suppose $\mathcal{B}$ is chosen to contain all the splitting primes with norm less than $L(\Delta)$. We make the following assumptions on Algorithm 2.4.2.

(a) Given a lattice $\mathcal{L}$, in a central simple algebra A' over $\mathbb{Q}$ with discriminant Δ' with Λ' a maximal order in it, and let $c > 0$ be a constant. We denote t as the smallest integer such that $\mathcal{L} \subseteq t^{-1}\Lambda'$, and define $N = t^{-n}[t^{-1}\Lambda' : L]^{1/n}$ (N is the reduced norm when L is a Λ'-ideal). Then there exists a constant $\alpha > 0$ such that when $\eta \in \mathcal{L}$ is any element with reduced norm less than $NL(\Delta')^{O(1)}$ in $\mathcal{L}$, the probability that the reduced norm of η is $\mathcal{B}$-smooth is at least $L(\Delta')^{-\alpha+o(1)}$.

(b) There is a constant ϵ, independent of the central simple algebra and the input ideal I, such that if $E \subseteq \mathrm{Mat}_2(B)$ is a set such that $[\langle\mathcal{B}\rangle : \langle N(E)\rangle]$ is finite, then each call of `LDivisibleElement` in step 4 in Algorithm 2.4.2 generates an output $x \in \mathrm{Mat}_2(B)$ such that $N(x)$ is not generated by $\langle N(E)\rangle$ with probability at least ϵ.

Part (a) of the heuristic is widely used in the `NextElement` procedure described in the rest of the paper. It is a reasonable because it can be thought of as a generalization of formula (1.16) in [Gra08], which describes the probability of $L(x)$-smooth numbers below x Part (b) of the heuristic involves the uniform distribution statements. It is natural in the sense that if we draw a random smooth element in $\langle \mathcal{B} \rangle$, we will expect it to be randomly distributed in the cosets of $\langle \mathcal{B} \rangle / \langle N(E) \rangle$.

Proposition 2.4.5. Assuming Heuristic 2.4.4 in addition. Algorithm 2.4.2 returns with X satisfying the conditions in Definition 2.4.1(ii) and terminates probabilistically in time $L(\Delta^{O(1)})$.

Proof. From the description of Definition 2.4.1, we know that it suffices to find a subset $E \in A^\times$ such that $\langle N(E) \rangle = \langle \mathcal{B} \rangle$. Indeed, if such E is found, for each $\ell \in \mathcal{B}$, we can just generate c_ℓ from E such that $N(c_\ell) = \ell$, as in steps 9 to 12 in the algorithm.

We know that if the loop between step 2 and 8 terminates, then we have $\langle N(E) \rangle = \langle \mathcal{B} \rangle$. So the question is whether it terminates. After step 1, we know that $[\langle \mathcal{B} \rangle : \langle N(E) \rangle] = 4^{|\mathcal{B}|}$, which is finite. Therefore, we only need finitely many essential updates of the set E between steps 5 and 7. And since the reduced norm map: $\Lambda \to \mathbb{Z}$ is surjective, the set E exists and the algorithm terminates. We shall show the complexity of the termination time based on Heuristic 2.4.4.

Now we make an estimate on the time complexity. We will first discuss the number of times `LDivisibleElement` is invoked, then discuss the complexity of `LDivisibleElement` routine in Algorithm 2.4.3.

The major part of the algorithm is the loop from step 2 to 8. As we assumed at Heuristics 2.4.4, the size of $\mathcal{B}$ is $L(\Delta)^{O(1)} / \ln L(\Delta)^{O(1)} = L(\Delta)^{O(1)}$, from the prime number theorem. At the first time the algorithm enters step 2, $[\langle \mathcal{B} \rangle : \langle N(E) \rangle] = 4^{L(\Delta)^{O(1)}}$. And each time when step 8 is entered and E is updated, the index $[\langle \mathcal{B} \rangle : \langle N(E) \rangle]$ is at least halved, so step 8 is executed $\log_2 4^{L(\Delta)^{O(1)}} = L(\Delta)^{O(1)}$ times. And, from Heuristics 2.4.4, we know that each random x generated by `LDivisibleElement`, from Heuristics 2.4.4, there is a possibility $\epsilon = O(1)$ that $N(x) \neq \langle N(E) \rangle$, so the number of times `LDivisibleElement` is invoked is also $O(1)L(\Delta)^{O(1)} = L(\Delta)^{O(1)}$.

It remains to discuss the complexity of `LDivisibleElement` routine in Algorithm 2.4.3, which involves finding a $\mathcal{B}$-smooth element by invoking the `NextElement` routine. By Heuristic 2.4.4(a), it takes in average $L(\Delta)^{O(1)}$ iterations to find a smooth element. Therefore, the complexity of `LDivisibeElement` is also $L(\Delta)^{O(1)}$. By taking product, the main loop between step 2 and step 8 is of time complexity $L(\Delta)^{O(1)}$.

Finally, step 10 only involves using the relations in $N(E)$ to simplify the elements in E using linear algebra, so the time should be polynomial in the size of the basis, that is $|\mathcal{B}| = L(\Delta)^{O(1)})$. Combining all the steps, the entire algorithm terminates in $L(\Delta)^{O(1)}$. □

2.4.2 The `GReduce` process

Given a general right Λ-ideal I, we need to first reduce it to a $\mathcal{B}$-smooth ideal by rescaling to $I' = sI$. It is usually preferable that the s we find have smaller entries, in the sense that $Q(s)$ is kept small. For this, one can take the naive approach, by invoking $s = \texttt{NextElement}(I^{-1})$ in Section 2.4.1 by a number of times, until one obtain a $\mathcal{B}$-smooth integral ideal of the form sI. From now on, we can assume that the input ideal I is $\mathcal{B}$-smooth.

Now we will describe the `GReduce` algorithm, which utilizes the G-reduction structure to simplify the structure of the input ideal. It also "extracts" two-sided ideals, in the sense that when we apply all the ℓ-reduction steps in Section 4, we will get the principal ideal generator (this will be justified in Theorem 2.6.1 in Section 2.6).

Algorithm 2.4.6 (The `GReduce` routine)**.**
Input: An integral $\mathcal{B}$-smooth right Λ-ideal I and a pre-computed G-reduction structure.
Output: A decomposition $fcI = J\mathfrak{J}$, where $f \in \mathbb{Q}^\times$, $c \in A^\times$, J is an integral right Λ-ideal not containing any proper two-sided integral ideals, and $\mathfrak{J}$ is a two-sided fraction Λ-ideal. Furthermore, $N(fcI) = 1$.

```
1: Compute and factorize $N(I)$. Assume that it is $\prod_{\ell \in \mathcal{B}} \ell^{-e_\ell}$.
2: Suppose $c_\ell$ is the element in $X$ which corresponds to $\ell \in \mathcal{B}$. Set $c \leftarrow \prod_{\ell \in \mathcal{B}} c_\ell^{e_\ell \pmod 4}$
   and $f \leftarrow \prod_{\ell \in \mathcal{B}} c_\ell^{\lfloor e_\ell / 4 \rfloor}$. Set $J \leftarrow cI$.
3: Let $\mathfrak{J}$ be the two-sided $\Lambda$-ideal generated by $J$, and $J \leftarrow J\mathfrak{J}^{-1}$.
4: return $f, c, J, \mathfrak{J}$.
```

Proposition 2.4.7. When the input in Algorithm 2.4.6 is a $\mathcal{B}$-smooth ideal with $\mathcal{B}$ chosen as in Heuristic 2.4.4, the outputs of Algorithm 2.4.6 are valid and return deterministically in time $L(\Delta)^{O(1)}$.

Proof. After step 2, since $N(f) = f^4$, it is clear that $N(fcI) = N(f)N(c)N(I) = 1$. After step 3, we have $cI = J\mathfrak{J}$, and since $\mathfrak{J}$ is the smallest two-sided Λ-ideal containing cI, $J = cI\mathfrak{J}^{-1}$ is does not contain a proper two-sided Λ-ideal and is integral.

For the complexity, since $N(I)$ is $\mathcal{B}$-smooth, finding the factorization in step 1 only involves trial divisions by elements in $\mathcal{B}$, which is of size $O(L(\Delta)^{O(1)} / \ln(L(\Delta)^{O(1)}) = O(L(\Delta)^{O(1)})$. Denote the number of prime divisors (with multiplicities) by $\Omega(N(I))$, which can be bounded by $\log_2 N(I)$, then the divisions we need to try in step 1 is $O(\ln N(I) + L(\Delta)^{O(1)})$. Step 2 is immediate from the output of Algorithm 2.4.2, and step 3 only involves a lattice basis generation on the generating set $\{lj\}_{l \in \Lambda, j \in J}$, which is polynomial time. Therefore, step 1 dominates the algorithm and takes $O(\ln N(I) + L(\Delta)^{O(1)})$ in time. □

2.5 The local reduction process

The goal of this section is to establish the `LReduce` algorithm as described in the main Algorithm 2.3.1, which solves the following question, which plays an important role in reducing the ℓ-adic part of an ideal:

Problem 2.5.1 (`LReduce`)**.** Let I be a right Λ-ideal, and $\ell \neq p$ be a prime splitting in B. Find an element $\gamma_\ell \in A^\times$, r a non-negative integer, and J another right Λ-ideal, such that $\gamma_\ell I = \ell^r J$, and $v_\ell(N(J))$, the valuation at ℓ of the reduced norm of J, is between 0 and 3.

We will propose an algorithm that generalizes Page's approach in [Pag14] to matrices of quaternions. The upshot of the algorithm for the local reduction process is the fact that the Bruhat-Tits building $\mathcal{T}_{\mathbb{Z}_\ell}$ on rank 4 $\mathbb{Z}_\ell$ lattices have a compatible Λ_ℓ-action on the right $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideal classes.

The main idea is to use the equivalence between the action of $A^\times$ on $\mathbb{Z}_\ell$-lattices of rank 4 and the action of $A^\times$ on right Λ_ℓ-ideals. The isomorphism classes of rank 4 $\mathbb{Z}_\ell$-lattices and their inclusion relations can be encoded as Bruhat-Tits buildings, and we discussed the theory and the $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-actions on the buildings in Section 2.2.4.

2.5.1 The compatibility between ideals and lattices actions

Now let B be a quaternion algebra over $\mathbb{Q}$, and $A = \mathrm{Mat}_2(B)$. Suppose $\mathcal{O}$ is a maximal order of B, and $\Lambda = \mathrm{Mat}_2(\mathcal{O})$. We know that Λ is a maximal order of A (see Section 2.2.3). Let ℓ be a prime in $\mathbb{Q}$ which splits in A. We denote $A_\ell := A \otimes_{\mathbb{Q}} \mathbb{Q}_\ell$ and similarly Λ_ℓ to be the completion of A and Λ at ℓ, respectively. As discussed in Section 2.2, we fixed an embedding $\iota : \Lambda \to \mathrm{Mat}_4(\mathbb{Z}_\ell)$, which extends to an isomorphism $\iota_\ell : A_\ell \xrightarrow{\sim} \mathrm{Mat}_4(\mathbb{Q}_\ell)$. We will identify $A_\ell = \mathrm{Mat}_2(B)_\ell$ with $\mathrm{Mat}_4(\mathbb{Q}_\ell)$ without mentioning the isomorphism ι_ℓ for brevity, if there is no confusion.

The traversal of the Bruhat-Tits buildings such as $\mathcal{T}_{\mathbb{Z}_\ell}$ plays a central role in our local reduction algorithm. In this subsection, we introduce the fact that in the local theory, there is a one-to-one correspondence between vertices in the Bruhat-Tits building (that is, homothety classes of $\mathbb{Z}_\ell$-lattices of rank 4) and the right $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideals modulo two-sided ideals. In addition, we can define a left action of elements in $\mathrm{GL}_4(\mathbb{Z}_\ell)$ on both objects, and the action is equivalent on the vertices of the Bruhat-Tits building and the ideals. We will use the theory we set up here to demonstrate in Section 2.5.5 that, when we simplify the lattice, we are also simplifying the corresponding ideal.

Proposition 2.5.2. Every right Λ_ℓ-ideal is principal.

Proof. See [Rei03, Theorem 18.7(ii)]. Note that this is true even if ℓ does not split in A. □

There is a natural action of $A_\ell^\times$ on right Λ_ℓ-ideals. Let $c \in A_\ell^\times$ and I be such an ideal. Then we simply take $c \cdot I = cI$. Back to our scenario, where $A = \mathrm{Mat}_2(B)$. Since our fixed embedding $\phi_2 : \Lambda \to \mathrm{Mat}_4(\mathbb{Z}_\ell)$ extends to an isomorphism $\phi_{2,\ell} : A_\ell \xrightarrow{\sim} \mathrm{Mat}_4(\mathbb{Q}_\ell)$, we can also define a natural action of $\mathrm{GL}_4(\mathbb{Q}_\ell)$ on right $\Lambda_\ell \cong \mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideals.

On the other hand, as described in Section 2.2.4, we have an action of $\mathrm{GL}_4(\mathbb{Z}_\ell)$ on lattices $P \subseteq \mathbb{Q}_\ell^4$, so that when $g \in \mathrm{GL}_4(\mathbb{Z}_\ell)$ and $P = \langle e_i \rangle_{i=1}^4$, $g \cdot P = \langle g \cdot e_i \rangle_{i=1}^4$. This action induces an action of $\mathrm{GL}_4(\mathbb{Z}_\ell)$ to $\mathcal{T}_{\mathbb{Z}_\ell}$.

In addition, there is a correspondence between $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideals and full lattices in $\mathbb{Q}_\ell^4$, by sending gI to $g \cdot P_0$. Among all the maps being set up, we have the following key observation.

Proposition 2.5.3. Suppose ℓ splits in A. The map $g \cdot P_0 \mapsto g \cdot \Lambda_\ell$ is bijective and equivariant between vertices in $\mathcal{T}_{\mathbb{Z}_\ell}$ and right-Λ_ℓ ideals modulo two-sided Λ_ℓ-ideals, under the action of $\mathrm{GL}_4(\mathbb{Q}_\ell)$ on the Bruhat-Tits building $\mathcal{T}_{\mathbb{Z}_\ell}$ and the action of $A_\ell \cong \mathrm{GL}_4(\mathbb{Q}_\ell)$ on right-Λ_ℓ ideals as described after Proposition 2.5.2.

Proof. The equivariance is clear, since $\alpha \cdot (g \cdot P_0)$ is mapped to $\alpha \cdot (g \cdot \Lambda_\ell)$ by definition. It remains to prove the bijection.

For any $g \in \mathrm{GL}_4(\mathbb{Q}_\ell)$, the stabilizer of the lattice $g \cdot P_0$ is given by Proposition 2.2.11 (ii), which is $g(\mathbb{Q}_\ell^\times \mathrm{GL}_4(\mathbb{Z}_\ell))g^{-1}$.

On the other hand, we want to find the stabilizer of the right Λ_ℓ-ideal $I_g := \iota_\ell^{-1}(g)\Lambda_\ell$. We know that $\alpha \cdot I_g = I_g$ if and only if $\alpha g^{-1}\Lambda_\ell = g^{-1}\Lambda_\ell$, or $g\alpha g^{-1} \in \Lambda_\ell^\times$. On the other hand, by [Rei03, Theorem 19.3], any two-sided $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideal in $\mathrm{Mat}_4(\mathbb{Q}_\ell)$ is generated by $\mathrm{rad}(\mathrm{Mat}_4(\mathbb{Z}_\ell)) = \ell \mathrm{Mat}_4(\mathbb{Z}_\ell)$. Therefore, considering I_g as an ideal class of right Λ_ℓ-ideal modulo a two-sided Λ_ℓ-ideal, the stabilizer is $g(\ell^{\mathbb{Z}}\mathrm{GL}_4(\mathbb{Z}_\ell))g^{-1} = g(\mathbb{Q}_\ell^\times \mathrm{GL}_4(\mathbb{Z}_\ell))g^{-1}$. The bijection follows because the stabilizers are the same. □

2.5.2 The ℓ-reduction structure: the definition

Throughout the remaining of Section 2.5, let $B = B_{p,\infty}$ be a quaternion algebra over $\mathbb{Q}$ with discriminant $\Delta = p$, $\mathcal{O}$ be a maximal order in B, $A = \mathrm{Mat}_2(B)$, and $\Lambda = \mathrm{Mat}_2(\mathcal{O})$. And let $\ell \neq p$ be a prime splitting in B. We have constructed in Section 2.2.4 a Bruhat-Tits building $\mathcal{T}_{\mathbb{Z}_\ell}$ of rank 4 $\mathbb{Z}_\ell$-lattices, together with the standard lattice P_0 which is considered as the center of $\mathcal{T}_{\mathbb{Z}_\ell}$, as in Definition 2.2.10(ii). Also, we fix the embedding $\iota : A \to \mathrm{Mat}_4(\mathbb{Q}_\ell)$.

Now we aim for an explicit method to simplify both the paths in the Bruhat-Tits building and the ideals, for which we will show later in Section 2.5.5 how to use it to answer Problem 2.5.1. More concretely, it is a consequence of Proposition 2.5.3 that a path containing a chamber in a Bruhat-Tits building corresponds to a right Λ-ideal which contains as a factor a two-sided ideal of norm ℓ^4, which turns out to be $\ell\mathrm{Mat}_4(\mathbb{Z}_\ell)$. Therefore, to extract the factor ℓ^r on the ideal side, the aim is to reduce a path in the Bruhat-Tits building and make them into cycles along a chamber. And for that, we need a pre-computed ℓ-reduction structure (as described in Definition 2.5.4) for each splitting prime ℓ of interest.

Our goal is to build up an extension of Page's method, which is for principal ideal problem over quaternion algebra. The main difference in the local side is that in quaternion algebra, completions of ideals will be ideals over $\mathrm{Mat}_2(\mathbb{Z}_\ell)$, which corresponds to a Bruhat-Tits tree. Compared to the Bruhat-Tits tree, we need the higher dimension analog, the Bruhat-Tits building, which is no longer a directed graph, and the building expanded in size much faster than the Bruhat-Tits tree as the prime ℓ grows. Such a difference results in the necessity to modify the definition of the ℓ-reduction structure.

Definition 2.5.4. Let $\ell \neq p$ as above. The ℓ-reduction structure for $\Lambda = \mathrm{Mat}_2(\mathcal{O})$ consists of the following data.

(i) A filtration of right Λ-ideals $\ell\Lambda \subsetneq M_3 \subsetneq M_2 \subsetneq M_1 \subsetneq \Lambda$, maximal orders $\Lambda_0 = \Lambda$, and $\Lambda_i = O_l(M_i)$ for $i = 1, 2, 3$. ℓ-adic generating global element g_i of M_i, such that in the ℓ-adic completion, $(M_i)_\ell = (g_i)_\ell\Lambda$. A chamber $([P_0], [P_1], [P_2], [P_3])$ satisfying $[P_i : P_{i+1}] = \ell$ for $i = 0, 1, 2$, and $[P_i]$ is the stabilizer of $\Lambda_i^\times$.

(ii) For each $i = 0, 1, 2, 3$: for each type 1 out-neighbor $[Q]$ of $[P_i]$, an element $c_Q \in \Lambda_i^\times$ such that $c_Q \cdot [Q] = [P_{i+1}]$. (Here we denote $[P_4] = [\ell P_0]$ for convenience).

The next step is to compute a ℓ-reduction structure. We first describe how to compute the data in part (i) in Algorithm 2.5.5 in Section 2.5.3. Algorithms for part (ii) will be discussed in Section 2.5.4. And we will show in Section 2.5.5 how to use the ℓ-reduction structure to complete the local reduction in Problem 2.5.1.

2.5.3 Computing the ℓ-reduction structure: finding the filtration of ideals and lattices

Here we will show how to compute the ℓ-reduction structure described in Definition 2.5.4(i).

Algorithm 2.5.5 (Maximal ideal decomposition of $\ell\Lambda$)**.**
Input: $\Lambda = \mathrm{Mat}_2(\mathcal{O})$, $\ell \neq p$ be a splitting prime. $\phi_1 : B \hookrightarrow \mathrm{Mat}_2(\mathbb{Q}_\ell)$, and $\phi_2 : \mathrm{Mat}_2(B) \hookrightarrow \mathrm{Mat}_4(\mathbb{Q}_l)$ is the embedding compatible to ϕ_1.
Output: Ideals $\{M_i\}_{i=1}^3$, maximal orders $\{\Lambda_i\}_{i=1}^3$, and the chamber $([P_0], [P_1], [P_2], [P_3])$.

```
1: $\mathfrak{P} \leftarrow$ a right $\mathcal{O}$-ideal of norm $\ell$.
```

```
2: $M_1 \leftarrow \begin{bmatrix} \mathcal{O} & \mathcal{O} \\ \mathfrak{P} & \mathfrak{P} \end{bmatrix}, M_2 \leftarrow \begin{bmatrix} \mathcal{O} & \mathcal{O} \\ \ell\mathcal{O} & \ell\mathcal{O} \end{bmatrix}, M_3 \leftarrow \begin{bmatrix} \mathfrak{P} & \mathfrak{P} \\ \ell\mathcal{O} & \ell\mathcal{O} \end{bmatrix}$.
3: $g_1 \leftarrow$ LadicGlobalGenerator$(M_1, \ell)$, $g_2 \leftarrow \begin{bmatrix} 1 & 0 \\ 0 & \ell \end{bmatrix}$,
       $g_3 \leftarrow$ LadicGlobalGenerator$(M_3, \ell)$.
           ▷ The routine LadicGlobalGenerator will be described in Algorithm 2.5.16.
4: $\mathcal{O}' \leftarrow$ LeftOrder$(\mathfrak{P})$.
5: $\Lambda_1 \leftarrow \begin{bmatrix} \mathcal{O} & \mathfrak{P}^{-1} \\ \mathfrak{P} & \mathcal{O}' \end{bmatrix}, \Lambda_2 \leftarrow \begin{bmatrix} \mathcal{O} & \ell^{-1}\mathcal{O} \\ \ell\mathcal{O} & \mathcal{O} \end{bmatrix}, \Lambda_3 \leftarrow \begin{bmatrix} \mathcal{O}' & \ell^{-1}\mathfrak{P} \\ \ell\mathfrak{P}^{-1} & \mathcal{O} \end{bmatrix}$.
6: $\mu_1 \leftarrow \phi_2(g_1), \mu_2 \leftarrow \phi_2(g_2), \mu_3 \leftarrow \phi_2(g_3)$.
7: $P_1 \leftarrow \mu_1 P_0, P_2 \leftarrow \mu_2 P_0, P_3 \leftarrow \mu_3 P_0$.
8: **return** $\{M_i\}_{i=1}^3, \{\Lambda_i\}_{i=1}^3, ([P_0], [P_1], [P_2], [P_3])$.
```

To ensure the algorithm terminates in expected time, we will need the following uniform distribution assumption.

Heuristic 2.5.6. Let I be a right $\mathrm{Mat}_4(\mathbb{Q}_\ell)$-ideal, and $v_\ell(N(I)) = \ell^N$. Then there is a constant ϵ, independent of the central simple algebra and the input ideal I, such that the map $I \to \mathbb{F}_\ell$, defined as the composition of the map $I \to \mathbb{Z}_\ell, x \to \ell^{-N}N(x)$ and the residue map, sends any element to zero with probability at least ϵ.

The assumption makes sense since we are indeed expecting a stronger statement, saying that the map $I \to \mathbb{F}_\ell$ should send I to all ℓ elements in $\mathbb{F}_\ell$ with equal probability.

Proposition 2.5.7. Algorithm 2.5.5 generates the correct outputs, probabilistically in polynomial time in the bit length of Δ, under heursitic 2.5.6.

Proof. After step 2, M_i clearly gives a filtration as $\mathbb{Z}$-modules between Λ and $\ell\Lambda$. Noticing that $\mathfrak{P}$ are right $\mathcal{O}$-ideals, it can be readily checked that $M_i\alpha \subseteq M_i$ for all i's and all $\alpha \in \Lambda$, therefore $\mathcal{O}_r(M_i)$ contains Λ and hence is Λ, from the maximality of Λ. For step 5, one can also check directly that $\Lambda_i M_i \subset M_i$. And since Λ_i are maximal orders (which follows from [Rei03, corollary 27.6]), Λ_i is the right order of M_i. Finally, for step 7, the ideals $\tilde{M}_i$ generated by $\phi_2(M_i)$ gives a filtration from $\mathrm{Mat}_4(\mathbb{Z}_\ell)$ to $\ell\mathrm{Mat}_4(\mathbb{Z}_\ell)$, and by comparing the norms, we see that the ℓ-adic generating global elements g_i are mapped by ϕ_2 to μ_i, which are ideal generators of $\tilde{M}_i$. From the compatibility in Section 2.5.1, we know that the lattices P_i satisfies the index and stabilizer properties as in Definition 2.5.4.

For the complexity, since most of Algorithm 2.5.5 only involves direct assignments and standard algorithms in quaternion algebra (finding left orders and inverse ideals), which are all polynomial time in Δ. The only non-standard part is the LadicGlobalGenerator routine, which will be described in Algorithm 2.5.16 in Section 2.5.5, involved in step 3. From Heuristic 2.5.6, we assume that the probability exiting the loop in steps 2 to 4 in Algorithm 2.5.16 is ϵ, which is a constant. Therefore, `LadicGlobalGenerator` also terminates in polynomial time in Δ, which implies the entire algorithm is also polynomial time in Δ. □

2.5.4 Finding transitive actions in the chamber

In Section 2.5.2, we defined in Definition 2.5.4 the data contained in the ℓ-reduction structure and gave an algorithm computing part (i) in Algorithm 2.5.5. Now we will focus on part (ii) of the definition. We see that finding a transitive action on $[P_0]$ is simpler than that on $[P_i]$ for $i = 1, \cdots, 3$, since the underlying left order Λ has a simpler form. An algorithm is described in Algorithm 2.5.9 in Section 2.5.4.1. For the rest of the transitive actions, we need a reduction to the $[P_0]$ case, where the details are described in Algorithm 2.5.14 in Section 2.5.4.2.

2.5.4.1 Computing the ℓ-reduction structure: finding transitive actions on $[P_0]$

To find a transitive action, we need a convenient way to describe the lattices and their neighbors. The neighbors of a lattice can be described by its kernel. With such an idea, it turns out that out-neighbors of type 1 edges of $[P_0]$ (or any other lattice) is bijective to the 3-dimensional projective space $\mathbb{P}^3(\mathbb{F}_\ell)$ over the finite field $\mathbb{F}_\ell$.

More precisely, suppose P' is a lattice such that $[P_0 : P'] = \ell$. Let $g \in \mathrm{Mat}_4(\mathbb{Z}_\ell)$ be a basis of P', i.e., $g \cdot P_0 = P'$. Let $\gamma \in \mathrm{Mat}_4(\mathbb{F}_\ell)$ be the image of g. Define $\tau \in \mathbb{F}_\ell^4$ as a column vector satisfying $\gamma^t \cdot \tau = 0$. Since such a τ is unique up to a multiplication in $\mathbb{F}_\ell^\times$, τ induces an element in $\mathbb{P}^3(\mathbb{F}_\ell)$.

Now we represented out vertices of $[P_0]$ as elements in $\mathbb{P}^3(\mathbb{F}_\ell)$ in the form of column vectors. Therefore, when $\Gamma = \mathrm{Mat}_4(\mathbb{F}_\ell), \mathrm{Mat}_4(\mathbb{Z}_\ell)$, or Λ, there is a natural mapping $\Gamma \to \mathrm{Mat}_4(\mathbb{F}_\ell)$ so that Γ acts on $\mathbb{P}^3(\mathbb{F}_\ell)$ by left multiplication. The following lemma relates actions on the out edges of $[P_0]$ and actions on $\mathbb{P}^3(\mathbb{F}_\ell)$.

Lemma 2.5.8. Let $\alpha \in \Lambda_\ell^\times \cong \mathrm{GL}_4(\mathbb{Z}_\ell)$. Then the action of α on $\mathbb{P}^3(\mathbb{F}_\ell)$ as the left multiplication of $\tilde{\alpha}$ on the column vector in $\mathbb{F}_\ell^4$ (here $\tilde{\alpha}$ is the reduction of α to $\mathrm{GL}_4(\mathbb{F}_\ell)$) is equivalent to the action of α^* on the type-1 out-neighbors of $[P_0]$, where $\alpha^* := (\alpha^{-1})^t$ is the contragradient of α.

Proof. Following the notations above, we first show that the bijection from the sublattices of $[P_0]$ to $\mathbb{P}^3(\mathbb{F}_\ell)$ is well-defined. Given a sublattice P' such that $[P_0 : P'] = \ell$, the choice of $g \in \mathrm{Mat}_4(\mathbb{Z}_\ell)$ is equivalent to the choice of a basis of P_0, and it is up to a right multiplication on $\mathrm{GL}_4(\mathbb{Z}_\ell)$. After the reduction of g, $\gamma \in \mathrm{Mat}_4(\mathbb{F}_\ell)$ is unique up to a right reduction of $\mathrm{GL}_4(\mathbb{F}_p)$ and have rank 3. Therefore τ is also uniquely defined up to a multiplication of $\mathbb{F}_p^\times$.

Now, assume $\alpha \in \mathrm{Mat}_4(\mathbb{Z}_\ell)$. Since $(\alpha_1\alpha_2)^* = \alpha_1^*\alpha_2^*$ and $(\alpha_1^{-1})^* = (\alpha_1^*)^{-1}$, we only need to prove that when P' is a sublattice of P_0 of index ℓ and $g \in \mathrm{Mat}_4(\mathbb{Z}_\ell), \tau \in \mathbb{F}_\ell^4$ correspond to P', then $\alpha\tau$ is mapped to $\alpha^* \cdot P'$. This is true since $(\alpha^* g)^t(\alpha\tau) = g^t(\alpha^*)^t\alpha\tau = g^t\tau = 0$. □

We will now use Lemma 2.5.8 to find a transitive action on $\mathbb{P}^3(\mathbb{F}_\ell)$. An important feature is, it is hard in general (at least subexponential time in terms of the discriminant of the quaternion algebra) to find a general unit in Λ, but there are subgroups in $\Lambda = \mathrm{Mat}_2(\mathcal{O})$ where reduced norm are easy to compute and unit elements are easy to find: $\mathrm{SL}_2(\mathbb{Z})$ and $H := \{\alpha \in A \mid \alpha = \alpha^*\}$, the Hermitian matrices. We will prove in Proposition 2.5.12 that $\mathrm{SL}_2(\mathbb{Z})$ and H generates a transitive action on the index ℓ sublattices of $[P_0]$.

Algorithm 2.5.9 (The routine `TransitiveAction`$([P], [P_0])$, which finds transitive actions on type 1 out-neighbors of $[P_0]$)**.**

Input: P, a sublattice of P_0 such that $[P_0 : P] = \ell$. Maps $\varphi_1 : B \to \mathrm{Mat}_2(\mathbb{F}_\ell)$, $\varphi_2 : \mathrm{Mat}_2(B) \to \mathrm{Mat}_4(\mathbb{F}_\ell)$ and pullback functions $\varphi_1^{-1}, \varphi_2^{-1}$ which returns elements in B and $\mathrm{Mat}_2(B)$, respectively, which has the smallest entries (in the sense that the quadratic form Q, as described in Section 2.4.1, is minimized) among the inverse image .

Output: A global element $g \in \Lambda^\times$ such that $g \cdot P = \mathrm{diag}(1,1,1,\ell) \cdot P_0$, where diag means the diagonal matrix with the prescribed entries.

```
$g \leftarrow \mathrm{Id}$.
$\tau \leftarrow$ SubLatticeToP3($P$). ▷ So $\tau$ is the column vector in $\mathbb{F}_p^4$ derived from the bijection.
if $\tau[1] \times \tau[4] = \tau[2] \times \tau[3]$ then
        ▷ This is the case when we cannot find a $g$ of the form $sh$, where $s \in \mathrm{SL}_2(\mathbb{Z})$ and $h \in H$.
        ▷ Instead, we attempt to find $g$ of the form $h_2 s_1 h_1$, where $s_1 \in \mathrm{SL}_2(\mathbb{Z}), h_1, h_2 \in H$.
    repeat
        $h_1 \leftarrow$ RandomElement(H), $\eta_1 \leftarrow \varphi_2(h_1)$
    until $(\eta_1^*\tau)[1] \times (\eta_1^*\tau)[4] \neq (\eta_1^*\tau)[2] \times (\eta_1^*\tau)[3]$
    $h_1 \leftarrow$ HermitianPullback($\eta_1$), $g \leftarrow h_1$, $\tau \leftarrow \eta_1^* \cdot \tau$.
end if
$\sigma_1 \leftarrow \begin{bmatrix} -\tau[4] & \tau[2] \\ -c\tau[3] & c\tau[1] \end{bmatrix}$, where $c \in \mathbb{F}_\ell$ satisfies $\det(\sigma_1) = 1$.
                                        ▷ After the action by $\sigma_1$, $\tau$ will take the form $[* \; 0 \; 0 \; *]^t$.
$s_1 \leftarrow$ SL2Pullback($\sigma_1^*$), $\tau \leftarrow \sigma_1 \cdot \tau$, $g \leftarrow s_1 \cdot g$. ▷ SL2Pullback returns a pullback $s_1$ in $\mathrm{SL}_2(\mathbb{Z})$.
```

```
11: $\eta_2 \leftarrow \begin{bmatrix} 1 & & & -\tau[4]^{-1}\tau[1] \\ & 1 & & \\ & \tau[4]^{-1}\tau[1] & u & \\ & & & u \end{bmatrix}$, where $u \in \mathbb{F}_\ell$ satisfies $\det(\eta_2) = 1$.
                                        ▷ After the action by $\eta_2$, $\tau$ will take the desired form $[0\ 0\ 0\ *]^t$.
12: $h_2 \leftarrow$ HermitianPullback($\eta_2^*$).        ▷ HermitianPullback returns a pullback of $\varphi_2$ in
    $H \subseteq \Lambda$.
13: $\tau \leftarrow \eta_2 \cdot \tau$, $g \leftarrow h_2 \cdot g$.
14: return $g$.
```

Remark 2.5.10. We sketch how the pullback routines `SL2Pullback` and `HermitianPullback` are obtained. We know that for $s' \in \mathrm{Mat}_2(\mathbb{Z})$, the reduced norm of s' in $\mathrm{Mat}_2(B)$ is simply $\det(s')^2$, `SL2Pullback`(σ) simply finds a matrix in $\sigma + \ell\mathrm{Mat}_2(\mathbb{Z})$ with determinant 1 and the entries are the smallest possible. Similarly, for a Hermitian matrix $h' = \begin{bmatrix} a & q \\ \bar{q} & d \end{bmatrix} \in H$, we have $N(h') = (N(q) - ad)^2$, where $N(q)$ is the reduced norm of $q \in B$. So `HermitianPullback`(η) simply pulls back η to $\mathrm{Mat}_2(B)$ using the pullback function ϕ_2, and then adjust the pullback by $\ell\Lambda$ to ensure unit reduced norm.

To ensure that Algorithm 2.5.9 terminates in expect time, we will hope that the process of left multiplying by η_1 have some uniform distribution property. We will formulate them below.

Heuristic 2.5.11. There is an constant ϵ, independent of the central simple algebra A and the input ideal I, satisfying the following property. Let $\tau = [\tau_1\ \tau_2\ \tau_3\ \tau_4]^t \in \mathbb{F}_\ell^4$ satisfy $\tau_1\tau_4 = \tau_2\tau_3$, and $\sigma = \begin{bmatrix} 1 & 0 & a & b \\ 0 & 1 & c & d \\ d & -b & u & 0 \\ -c & a & 0 & u \end{bmatrix} \in \mathrm{GL}_4(\mathbb{F}_\ell)$ be a randomly chosen matrix. And let $\mu = [\mu_1\ \mu_2\ \mu_3\ \mu_4]^t$ be $\sigma\tau$. Then $\mu_1\mu_4 - \mu_2\mu_3$ equals to zero with probability at least ϵ.

Proposition 2.5.12. Algorithm 2.5.9 terminates with the correct output, probabilistically in polynomial time in the bit length of Δ, assuming Heuristic 2.5.11.

Proof. First, we need to show that the loop between line 4 and line 7 terminates. After the map $\phi_2 : \mathrm{Mat}_2(B) \to \mathrm{Mat}_4(\mathbb{F}_\ell)$, a Hermitian matrix in $h \in \mathrm{Mat}_2(B)$ with unit norm will be mapped to a matrix in $\mathrm{Mat}_4(\mathbb{F}_\ell)$ of the form $\eta = \varphi_2(h) = \begin{bmatrix} t & 0 & a & b \\ 0 & t & c & d \\ d & -b & u & 0 \\ -c & a & 0 & u \end{bmatrix}$, and

$\det(\eta) = (tu - ad + bc)^2 = 1$. The following statecment can be checked by splitting into cases and checking directly: for all $\tau = [\tau_1\ \tau_2\ \tau_3\ \tau_4]^t$ satisfying $\tau_1\tau_4 = \tau_2\tau_3$, there exists at least a matrix η of the above form so that $\eta \cdot [\tau_1\ \tau_2\ \tau_3\ \tau_4]^t = [\mu_1\ \mu_2\ \mu_3\ \mu_4]^t$ satisfies $\mu_1\mu_4 \neq \mu_2\mu_3$, which ensures the termination of the loop. (Indeed, we expected that a randomly chosen η gives approximately a probability $1 - 1/\ell$ to have $\mu_1\mu_4 \neq \mu_2\mu 3$ and terminates the loop.)

After step 8, we have $\tau = [\tau_1\ \tau_2\ \tau_3\ \tau_4]^t$ satisfying $\tau_1\tau_4 \neq \tau_2\tau_3$. On step 9, consider σ_1 as an element in $\mathrm{SL}_2(\mathbb{Z})$, $\varphi_2(\sigma_1) = \begin{bmatrix} -\tau_4 & 0 & \tau_2 & 0 \\ 0 & -\tau_4 & 0 & \tau_2 \\ -c\tau_3 & 0 & c\tau_1 & 0 \\ 0 & -c\tau_3 & 0 & c\tau_1 \end{bmatrix}$. It is an immediate check that $\det(\varphi_2(\sigma_1)) = 1$ and $\varphi_2(\sigma_1) \cdot \tau$ is of the form $[*\ 0\ 0\ *]^t$. Therefore, the pullback to $\mathrm{SL}_2(\mathbb{Z})$ exists, and we get an unit s_1 in $\Lambda^\times$ with the same action.

Finally, when $\tau_4 \neq 0$, the matrix η_2 in step 11 gives a column vector of the form $[0\ 0\ 0\ *]^t$ after the action on τ. And from Lemma 2.5.8, we know that embedding to $\mathrm{Mat}_4(\mathbb{Z}_\ell)$, η_2^* sends the corresponding sublattice to $\mathrm{diag}(1, 1, 1, \ell)P_0$. For the validity of the Hermitian pullback, if h is a Hermitian matrix, then $\varphi_2(h)$ is of the form $\begin{bmatrix} t & 0 & a & b \\ 0 & t & c & d \\ d & -b & u & 0 \\ -c & a & 0 & u \end{bmatrix}$ with contragradient $\begin{bmatrix} u & 0 & -d & c \\ 0 & u & b & -a \\ -a & -c & t & 0 \\ -b & -d & 0 & t \end{bmatrix}$. Although there is no straightforward formula for the pullback of the contragradient given h, we know from the explicit form that it is still a pullback from a unit norm Hermitian matrix in H. Therefore, the entire algorithm can be computed, terminates, and gives the correct output.

We will postpone the complexity statement and prove it in the more general situation in Proposition 2.5.13. □

At this point, we established Algorithm 2.5.9, which allows us to compute the transitive units as described in Definition 2.5.4(ii) for $i = 0$. Let $[Q]$ be a neighborhood of $[P_0]$, we can construct the routine `TransitiveAction`($[Q]$, $[P_0]$), which computes an global unit $c_Q \in \Lambda^\times$ such that $c_Q \cdot [Q] = [P_1]$ as follows. We input Q and $[P_1]$ to Algorithm 2.5.9, and suppose the outputs are g_Q and g_{P_1}, respectively. Then the routine `TransitiveAction`($[Q]$, $[P_0]$) returns $g_{P_1}^{-1} g_Q$.

2.5.4.2 Finding a transitive action on $[P_1]$ and beyond

We described an algorithm in Section 2.5.4.1 to generate a transitive action on neighbors of $[P_0]$. And as we will see in Section 2.5.5, we also need to generate a transitive action on neighbors of other elements in the chamber, namely $[P_1], [P_2]$, and $[P_3]$. The strategy we take is to translate the problems back to finding transitive action on neighbors of $[P_0]$.

As in Definition 2.5.4, we let $\Lambda_i = O_l(M_i)$ be the maximal order in the ℓ-reduction structure, $g_i \in \mathrm{Mat}_2(B)$ be an ℓ-adic generating global element of M_i, so that $(M_i)_\ell = (g_i)_\ell \Lambda$. And although we do not know how to compute this, we denote by $\tilde{g}_i \in M_i$ a left generator of the ideal M_i. In other words, $M_i = \tilde{g}_i \Lambda$ and $\Lambda_i = \tilde{g}_i \Lambda \tilde{g}_i^{-1}$. As in the previous sections, we also need to fix embeddings $\varphi_1 : B \hookrightarrow \mathrm{Mat}_2(\mathbb{Q}_\ell)$ and $\varphi_2 : \mathrm{Mat}_2(B) \hookrightarrow \mathrm{Mat}_4(\mathbb{Q}_\ell)$, but we omit them when there is no confusion in the notation.

Proposition 2.5.13. Consider the Eichler order $\Lambda' := g_i^{-1} \Lambda_i g_i \cap \Lambda \subseteq \Lambda$. Then finding a transitive action of $\Lambda'^{\times}$ on $[P_i]$ can be reduced to finding a transitive action of Λ on $[P_0]$.

Proof. Let $\tau := \tilde{g}_i^{-1} g_i$. Then we know that $\tau \in \Lambda$ and $\Lambda' = \tau^{-1} \Lambda \tau \cap \Lambda$ by definition. Suppose P is a lattice such that $[P_i : P] = \ell$. This means that there exists a $\mu \in \mathrm{Mat}_4(\mathbb{Z}_\ell)$ such that $v_\ell(\mu) = 1$ and $P = g_i \cdot \mu \cdot P_0$. Suppose that $g = g_i g' g_i^{-1}$ for some $g' \in \Lambda'$. From the definition, we know that $g \in \Lambda_i$ and $g' \in \Lambda$. Then we claim that such g acts transitively on the type 1 neighbors of $[P_i]$. Indeed, we have $g \cdot P = g_i g' g_i^{-1} g_i \mu P_0 = g_i g' \mu P_0$. Compare with Lemma 2.5.8, we see that the action of $g_i g' g_i^{-1}$ on P is equivalent to the action of g' on $\mu \cdot P_0$, which is a neighbor of P_0.

We will then prove that Λ' (which is a subset of Λ) contains sufficiently many units so that $g_i \Lambda'^{\times} g_i^{-1}$ generates a transitive action on neighbors of $[P_i]$, or equivalently, $\Lambda' \times$ generates a transitive action on neighbors of $[P_0]$. Even stronger, we will prove that Λ' contains sufficiently many elements in $\mathrm{SL}_2(\mathbb{Z})$ and Hermitian matrices (denoted by H again) which induces either transitive action.

We know that as $\mathbb{Z}$-modules, $\Lambda / \Lambda' \cong \oplus_p \Lambda_p / \Lambda'_p$, where p runs through all finite places in $\mathbb{Z}$. Therefore, the index $[\Lambda : \Lambda']$ can be obtained by multiply all the indices $[\Lambda_p : \Lambda'_p]$. We also know that the normalizer $\mathcal{N}_{A_\ell}(\Lambda_\ell)$, defined as $\{\sigma \in A_\ell^\times \mid \sigma \Lambda_\ell \sigma^{-1} = \Lambda_\ell\}$, is $\mathbb{Q}_\ell^\times \Lambda_\ell^\times$. From the definitions of g_i and $\tilde{g}_i$, we know that $v_\ell(N(g_i)) = v_\ell(N(\tilde{g}_i))$, so $\tau = \tilde{g}_i^{-1} g_i \in \Lambda_\ell^\times$ and falls in $\mathcal{N}_{A_\ell}(\Lambda_\ell)$. This implies that $\Lambda'_\ell = (\tau^{-1} \Lambda \tau \cap \Lambda)_\ell = (\tau^{-1} \Lambda \tau)_\ell \cap \Lambda_\ell = \Lambda_\ell$. Therefore, $\ell \nmid [\Lambda : \Lambda']$.

Now, suppose $n = [\Lambda : \Lambda']$. As $1 \in \Lambda'$, $\Lambda' \supseteq \mathbb{Z} + n\Lambda$. From the proof of Proposition 2.5.12, given any lattice P_0 such that $[P : P_0] = \ell$, there exist elements $\sigma_1 \in \mathrm{SL}_2(\mathbb{Z}) / \ell \mathrm{SL}_2(\mathbb{Z})$ and $\eta_1, \eta_2 \in H / \ell H$, such that whenever $s_1 \in \sigma_1 + \ell S_2(\mathbb{Z})$ and $h_i \in \eta_i + \ell H$, $i = 1, 2$ are units in Λ, $(h_2 s_1 h_1)^* \cdot P = \mathrm{diag}(1, 1, 1, \ell) \cdot P_0$. Since $\ell \nmid n$, by Chinese remainder theorem, is is possible to choose s_1, h_1 and h_2 such that they are units in Λ'. □

The consequence of Proposition 2.5.13 is that we can tweak the structure of Algorithm 2.5.9 to construct an algorithm to generate transitive action on neighborhoods of $[P_i]$ for $i = 1, 2$ and 3. We describe the algorithm as below.

Algorithm 2.5.14 (Translating the problem of finding the transitive action on Λ_i on the out-edges of $[P_i]$ to the problem of finding the transitive action on $\Lambda' \subseteq \Lambda$ to $[P_i]$)**.**

Input: All the local data as described in Definition 2.5.4. $i \in \{1, 2, 3\}$. P, a sublattice of P_i such that $[P_i : P] = \ell$. Maps $\varphi_1 : B \to \mathrm{Mat}_2(\mathbb{F}_\ell)$, $\varphi_2 : \mathrm{Mat}_2(B) \to \mathrm{Mat}_4(\mathbb{F}_\ell)$ and pullback functions $\varphi_1^{-1}, \varphi_2^{-1}$ which returns the element in B and $\mathrm{Mat}_2(B)$, respectively in the inverse image which has the smallest entries (in the sense that the quadratic form Q, as described in Section 3.1, is minimized) among the inverse image.

Output: A global element $g \in \Lambda_i^\times$ such that $g \cdot P = g_i \cdot \mathrm{diag}(1, 1, 1, \ell) P_0$.

```
$g \leftarrow \mathrm{Id}$.
$\psi \leftarrow$ SubLatticeToP3$(g_i^{-1} P)$.
        ▷ $\psi$ is the column vector in $\mathbb{F}_p^4$ corresponding to the sublattice $g_i^{-1} \cdot P$ of $P_0$ of index $\ell$.
$\Lambda' \leftarrow g_i^{-1} \Lambda_i g_i \cap \Lambda$, and $n \leftarrow [\Lambda : \Lambda']$.
if $\psi[1] \times \psi[4] = \psi[2] \times \psi[3]$ then
        ▷ This is the case when we cannot find a $g$ of the form $sh$, where $s \in \mathrm{SL}_2(\mathbb{Z})$ and $h \in H$.
        ▷ Instead, we try to find a $g$ of the from $h_2 s h_1$, where $s \in \mathrm{SL}_2(\mathbb{Z})$, and $h_1, h_2 \in H$.
    repeat
        $h_1 \leftarrow$ RandomElementModN$(H, n))$ $\eta_1 \leftarrow \varphi_2(h_1)$
                ▷ RandomElementModN$(H, n))$ gives a Hermitian element of the form $\mathrm{diag}(1, 1) + n\Lambda$.
    until $(\eta_1^* \psi)[1] \times (\eta_1^* \psi)[4] \neq (\eta_1^* \psi)[2] \times (\eta_1^* \psi)[3]$
    $g \leftarrow h_1$, $\psi \leftarrow \eta_1^* \cdot \psi$.
end if
$\sigma_1 \leftarrow \begin{bmatrix} -\psi[4] & \psi[2] \\ -c\psi[3] & c\psi[1] \end{bmatrix}$, where $c \in \mathbb{F}_\ell$ satisfies $\det(\sigma_1) = 1$.
$s_1 \leftarrow$ SL2PullbackModN$(\sigma_1^*, n)$, $\psi \leftarrow \sigma_1 \cdot \psi$, $g \leftarrow s_1 \cdot g$.
        ▷ SL2PullbackModN gives a pullback in $s_1 \in \mathrm{SL}_2(\mathbb{Z})$ of $\sigma_1$, congruent to $\mathrm{diag}(1, 1)$ mod $n$.
```

```
12: $\eta_2 \leftarrow \begin{bmatrix} 1 & & & -\psi[4]^{-1}\psi[1] \\ & 1 & & \\ & \psi[4]^{-1}\psi[1] & u & \\ & & & u \end{bmatrix}$, where $u \in \mathbb{F}_\ell$ satisfies $\det(\eta_2) = 1$.
13: $h_2 \leftarrow$ HermitianPullbackModN($\eta_2^*, n$).
        ▷ Gives a pullback $h_2 \in H$ of $\eta_2$ which is in $\mathrm{diag}(1,1) + n\Lambda$.
14: $\psi \leftarrow \eta_2 \cdot \psi$, $g \leftarrow h_2 \cdot g$.
15: return $g_1 \cdot g \cdot g_1^{-1}$.
```

Proposition 2.5.15. Algorithm 2.5.14 terminates with the correct output, probabilistically in polynomial time in Δ, assuming Heuristic 2.5.11.

Proof. We have already shown in Proposition 2.5.13 that for each $i \in \{1,2,3\}$, the action of $g_i g' g_i^{-1}$ on P is equivalent to the action of g' on $g_i^{-1}P$, which is a neighbor of P_0, and this proves the validity of steps 2 and 15. We also verified in Proposition 2.5.13 that Λ' contains enough units to generate the transitive action on neighbors of $[P_0]$. This guarantees the existence of the pullbacks in steps 8, 11, and 13. Finally, since the choice of matices η_1, σ_1 and η_2 in steps 6, 10 and 12 are the same as those in Algorithm 2.5.9, we know from Proposition 2.5.12 that the composition of actions of h_1, s_1, and h_2 will bring the lattice $g_i^{-1}P$ back to $\mathrm{diag}(1,1,1,\ell)P_0$.

For the complexity part, since Algorithm 2.5.9 is simpler than Algorithm 2.5.14, we will only check the later one. The first two steps are trivial, and step 3 involves a lattice basis computation, which is polynomial time in Δ. For the loop in steps 4 to 9, from the assumption in Heuristic 2.5.11, we know that it involves $O(1)$ calls to step 6, so it is probabilistically polynomial time.

The rest of the algorithm involves constantly many finite fields and integral arithmetic and Chinese remainder theorems, and they are all deterministic polynomial time. □

Now, we can finish up the routine `TransitiveAction`($[Q], [P_i]$) for $i \in \{1,2,3\}$, which inputs $[Q]$, a type 1 out neighbor of $[P_i]$, and outputs a global unit $c_Q \in \Lambda_i^\times$ satisfying $c_Q \cdot [Q] = [P_{i+1}]$. We proceed by invoking alrogithm 2.5.14 twice, taking Q and P_{i+1} as inputs. Suppose the outputs are g_Q and $g_{P_{i+1}}$, then the routine `TransitiveAction`($[Q], [P_i]$) returns $g_{P_{i+1}}^{-1} g_Q$.

2.5.5 The `LReduce` algorithm

Up to this point, we should have all components of the ℓ-reduction structure prepared, as described in Definition 2.5.4. We can finally describe Algorithm 2.5.18, which utilizes the ℓ-reduction data to traverse the Bruhat-Tits building. This is where we are using the compatibility in Section 2.5.1: a filtration of lattices in $\mathbb{Z}_\ell$ corresponds to a filtration of right $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideals, and the lattice action is equivalent to the ideal actions. So if we act on the lattice P by $\gamma \in \mathrm{Mat}_4(\mathbb{Z}_\ell)$ such that the filtration of γP consists of chambers in the Bruhat-Tits building, the action by γ on the corresponding ideal I, namely, γI will have two-sided ideal factors, which must be ℓ^r.

Since we need to exploit the correspondence between ideals and lattices, the first step is to map the given right Λ-ideal I to the corresponding $\mathbb{Z}_\ell$-lattice of rank 4. Indeed, if we know an element $\lambda \in I$ such that $I_\ell = \lambda\Lambda_\ell = \lambda\mathrm{Mat}_4(\mathbb{Z}_\ell)$, then we can correspond the ideal I with the lattice $\lambda_\ell \cdot P_0$. We call such an element λ a ℓ-adic generating global element.

Therefore, our first task is to find a ℓ-adic generating global element for the given right-Λ ideal I. This can be achieved by the following probabilistic algorithm.

Algorithm 2.5.16 (The `LadicGlobalGenerator` routine: computing a ℓ-adic generating global element)**.**
Input: An right Λ-ideal I. A splitting prime ideal ℓ in $\mathbb{Q}$.
Output: A ℓ-generating global element λ.

```
$\lambda \leftarrow 0$
while $v_\ell(N(\lambda)) \neq v_\ell(N(I))$ do
    $\lambda \leftarrow$ NextElement($I$).        ▷ See Section 3.1 for the routine NextElement.
end while
return $\lambda$.
```

Validity of Algorithm 2.5.16. From [Rei03, Theorem 24.2(a)], we know that $N(I)_\ell = N(I_\ell)$, and both of them will be the same as $\min_{c\in I} N(c)$. Therefore, if $v_\ell(N(\lambda)) = v_\ell(N(I))$ holds, λ is the ℓ-adic generating global element of I. Heuristically, a randomly chosen λ has probability $1 - 1/\ell$ to attain the minimal valuation an become an ℓ-adic generating global element (we stated this in Heuristic 2.5.6). Therefore, for any $\epsilon > 0$, the probabilistic approach terminates in $O(1)$ iterations in the loop with probability $1 - \epsilon$. □

Remark 2.5.17. Page has a deterministic algorithm in the quaternion algebra case, using a generalized Euclid's algorithm on the matrix ring $\mathrm{Mat}_2(\mathcal{O}_F/\mathfrak{p}^k)$.

Now we are ready to explain the `LReduce` algorithm. First, we will write down the algorithm as Algorithm 2.5.18, then we will illustrate how to simplify the lattice to reduce it as cycles in the Bruhat-Tits building. Finally, the validity is proven in Proposition 2.5.19.

Algorithm 2.5.18 (The `LReduce` routine)**.**
Input: An integral right Λ-ideal I, a prime $\ell \neq p$, splitting in B, the ℓ-reduction structure as described in Definition 2.5.4.

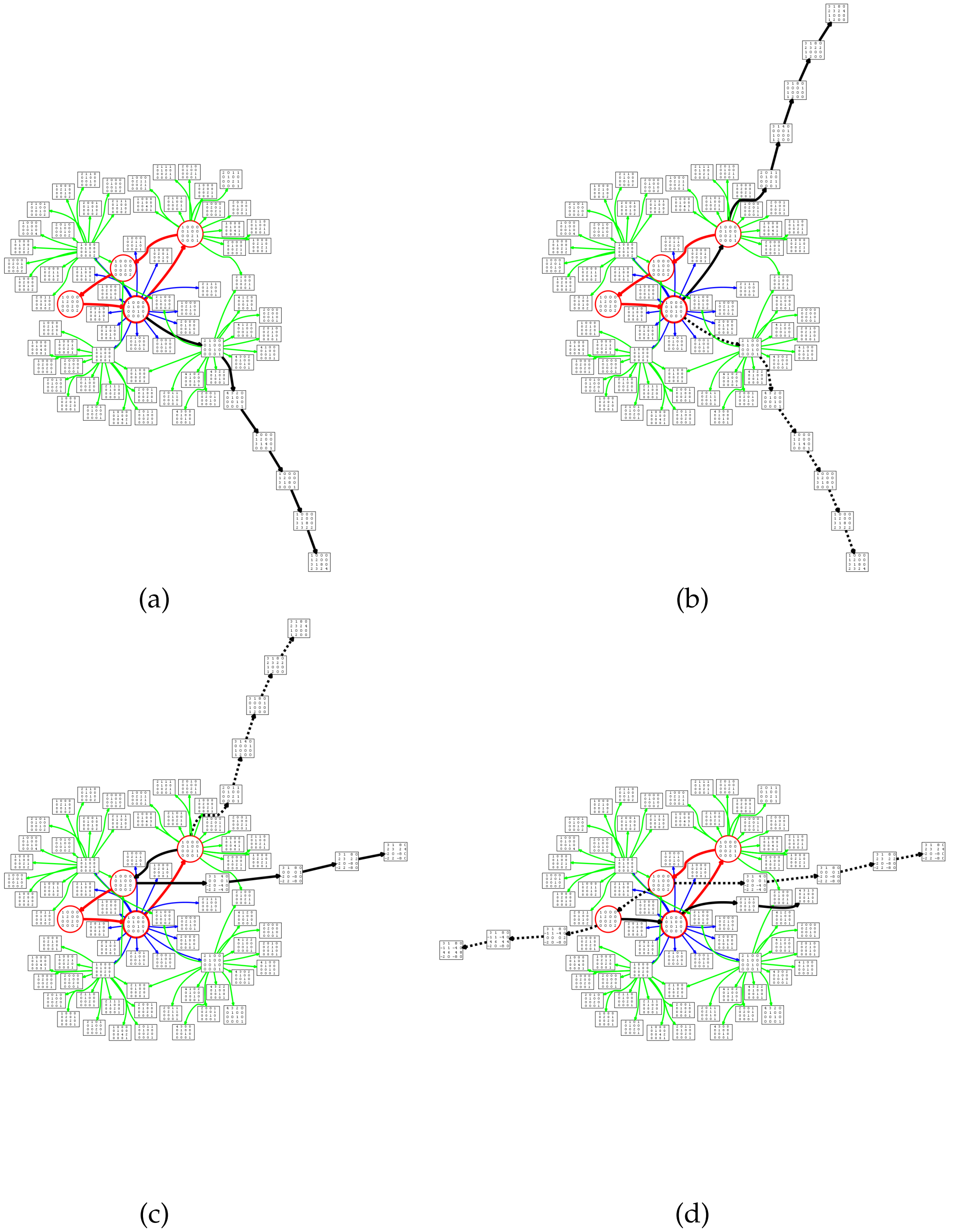
Output: An element $\gamma_\ell \in A^\times$, a non-negative integer r, a right Λ-ideal J, such that $\gamma_\ell I = \ell^r J\mathrm{m}$, and $v_\ell(N(J))$, the ℓ-adic valuation of the reduced norm of the ideal J, is between 0 and 3.

```
$\gamma_\ell \leftarrow 1, r \leftarrow 0, d \leftarrow v_\ell(N(I)), J \leftarrow I$.
$\lambda \leftarrow$ LadicGlobalGenerator$(I, \ell)$, $Q_J \leftarrow \lambda \cdot P_0$.
Compute a filtration of $\mathbb{Z}_\ell$-lattices $P_0 = L_0 \supset L_1 \supset \cdots \supset L_d = Q_J$, so that for each $0 \le i \le d-1$, $[L_i : L_{i+1}] = \ell$.
while $d > 3$ do
        ▷ Outer loop of the traversal of Bruhat-Tits building.
        ▷ Will try to make as many chambers as possible.
    for $i = 0$ to 3 do
            ▷ The inner loop of the traversal.
            ▷ For every four actions applied on the loop, a chamber will be formed on the filtration.
        $c_i \leftarrow$ TransitiveAction$(L_{4r+i+1}, P_i)$.    ▷ So $c_i \in \Lambda_i^\times$ satisfies $c_i \cdot L_{4r+i+1} = P_{i+1}$.
        $L_{4r+i+2} \leftarrow c_i \cdot L_{4r+i+2}, \cdots, L_{4r+d} \leftarrow c_i \cdot L_{4r+d}$.
    end for
    $\gamma_\ell \leftarrow c_3c_2c_1c_0\gamma_\ell, r \leftarrow r+1, d \leftarrow d-4, J \leftarrow \ell^{-1}\gamma_\ell J$.
    $L_{4r} \leftarrow \ell^{-1} \cdot L_{4r}, \cdots, L_{4r+d} \leftarrow \ell^{-1} \cdot L_{4r+d}$.
end while
return $\gamma_\ell, r, J$.
```

Here we explain how Algorithm 2.5.18 traverses the Bruhat-Tits building. Figure 2.1 illustrates an example when $\ell = 2$, that is, the Bruhat-Tits tree of rank 4 $\mathbb{Q}_\ell$-lattices.

We first describe the general setting for Figure 2.1. The Bruhat-Tits building is an infinite graph, but we only show the part around the center $[P_0]$ for simplicity. The vertices of the building are the homothety classes, and we represent them as matrices, with column vectors representing a basis of a lattice inside the homothety class. The matrices are written in the nodes in Figure 2.1. The central homothety class of the Bruhat-Tits building, as defined in Definition 2.2.10, is the bold red node. The fixed chamber, $([P_0], [P_1], [P_2], [P_3])$, in the Bruhat-Tits building as required in Definition 2.5.4(a) is the cycle with red edges in Figure 2.1. In addition to Figure 2.1, all the vertices which are of distance 1 from the origin (which are the heads of the blue arrows) and a part of the vertices which are of distance from the origin (which are the heads of the green arrows) are drawn.

Suppose we are given an ideal as an input in Algorithm 2.5.18. The first step is to find the corresponding $\mathbf{Z}_\ell$-lattice and a filtration of the lattice, which gives a path, as drawn in bold black edges in figure 2.1(a). In this example, the filtration has length $d = 6$, and



(a) (b) (c) (d)

Figure 2.1: The traversal of the Bruhat-Tits building of for $\ell = 2$, as in Algorithm 2.5.18. (a) The filtration of the lattice (in bold black edges) and the Bruhat-Tits building. The central lattice $[P_0]$ is in bold red, and the chamber $([P_0], [P_1], [P_2], [P_3])$ is the red loop. (b) The reduction after the first iteration in step 5 in Algorithm 2.5.18. The black bold path shows the updated filtration. (c) The reduction after the second iteration. (d) The reduction after the third and the fourth iterations.

suppose it is $P_0 = L_0 \supset L_1 \cdots \supset L_6$.

At the first step of the inner loop (step 5 of Algorithm 2.5.18, $i = 0$), we need to find an element $c_0 \in \Lambda_0^\times$ which sends L_1 to P_1. After acting c_0 on the entire filtration, we get a new filtration $P_0 = L_0' \supset P_1 = L_1' \supset L_2' \cdots \supset L_6'$. The new filtration is shown in the bold path in Figure 2.1(b).

Next, we will try to find an element in $c_1 \in \Lambda_1^\times$, which sends L_1' to P_2. Furthermore c_1 sends the filtration to $P_1 = L_1'' \supset P_2 = L_2'' \supset L_3'' \cdots L_6''$, as shown in the bold path of Figure 2.1(c).

Repeat the same process, and find $c_2 \in \Lambda_2^\times, c_3 \in \Lambda_3^\times$, and act each of them on the filtration. The filtration after action by c_3 is shown in Figure 2.1(d). After a complete round of traversal (four actions), the length of the filtration is shortened by four. And if we do the same action of the corresponding ideal I, we claim that a two-sided factor ℓ can be extracted from $c_3c_2c_1c_0I$, which consequently simplifies the ideal.

Now we shall prove the validity of the process in Proposition 2.5.19.

Proposition 2.5.19. Algorithm 2.5.18 terminates and outputs γ_ℓ, r, J with the desired property: $\gamma_\ell I = \ell^r J$, and it terminates probabilistically in polynomial time in the bit lengths of Δ and $N(I)$, assuming Heuristic 2.5.6.

Proof. It is clear that after an iteration in the while loop, d is decreased by 4, which is the only place where it is altered. Therefore, the algorithm terminates with finitely many while loops.

For the validity of the algorithm, we need to keep track of the correspondence between right $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideals and $\mathbb{Z}_\ell$ lattices of rank 4. After the setup in step 2 of Algorithm 2.5.18, we know that $\gamma_\ell I_\ell = I_\ell = \lambda\Lambda_\ell$ and $L_{4r+d} = Q_J = \lambda \cdot P_0$, so they are compatible in the sense of Proposition 2.5.3. After each iteration in the while loop, since $\gamma_\ell I_\ell$ and L_{4r+d} are both changed by the action of $c_3c_3c_1c_0$, so they remain compatible throughout the algorithm.

In each iteration of the while loop, the lattices $L_0, \cdots, L_{4r+d}$ remains to form a path of type 1 edges inside the Bruhat-Tits building $\mathcal{T}_\ell$. However, after each iteration, since $L_{4r+i} = P_i$ for $i = 1, 2, 3, 4$, the path formed by $\{L_i\}$ forms a new cycle of length 4. Therefore, after extracting an ℓ in step 10, the lattices $\{L_i\}$ remains to be a sublattice of P_0.

And at the end of the algorithm, the distance between P_0 and L_{4r+d} in the Bruhat-Tits building is at most 3. From Proposition 2.5.3, there is an integral $\mathrm{Mat}_4(\mathbb{Z}_\ell)$-ideal J_ℓ of norm ℓ^d such that both it and $\gamma_\ell I_\ell$ correspond to the lattice L_{4r+d}, and they differ by a two-sided $\mathrm{Mat}_4(\mathbb{Z}_\ell)$ ideal, which is $\ell^r\mathrm{Mat}_4(\mathbb{Z}_\ell)$ by comparing the exponents. And since γ_ℓ is a product of units in Λ_i for $i \in \{0, 1, 2, 3\}$, we know that at any completion $\ell' \neq \ell$ (including the ramified places), $(\ell^{-r}\gamma_\ell I)_{\ell'}$ is integral in $\Lambda_{\ell'}$. And from the previous argument, at the place ℓ, $(\ell^{-r}\gamma_\ell I)_\ell = J_\ell$ is also integral. Hence, $\ell^{-r}\gamma_\ell I = J$ is also an integral ideal since it is integral at every completion. This justifies the correctness of our output.

For the complexity, the first nontrivial step is step 2, which computes the ℓ-adic generatic global element, and takes probabilistic polynomial time under the assumption in Heuristic 2.5.6. Computing the filtration in step 3 can be implemented in probabilistic polynomial time, for instance, by sequentially inserting elements into the lattice Q_J until the filtration is fine enough in the sense that every two neighboring lattices are of index ℓ.

Next, we look at the loop from steps 4 to 11. The number of iteration we need depends on $v_\ell(N(I))$, which is polynomial in bit length of $N(I)$. And in each iteration, we need two invocations of Algorithm 2.5.14 or Algorithm 2.5.9, which is known to be polynomial from Proposition 2.5.13. The remaining are standard lattice and ideal operations, which take polynomial time. □

2.6 Putting everything together: the validity and the complexity analysis

So far we have described and showed the validity and complexity of all the components of our algorithm. Now we will combine everything together and discuss the validity and complexity. The heuristics and complexity settings are the same as the previous sections: for a real number x, we let $L(x)$ be the function $L(x) = \exp((\ln x \ln\ln x)^{1/2})$. Suppose $\mathcal{B}$ is the set of primes in $\mathbb{Q}$ which are less than $L(\Delta)^{O(1)}$ and not equal to p (i.e., splits in B). Again, we list a few heuristic assumptions which are needed for the size and complexity estimate of Theorem 2.6.1.

Theorem 2.6.1. The main algorithm (Algorithm 2.3.1) is valid and terminates probabilistically in time $L(\Delta)^{O(1)}$. In addition, the output of the algorithm is a generator of $I \subseteq \mathrm{Mat}_2(B)$ whose bit lengths of entries are $O(\ln(N(I)) + \ln(L(\Delta)))$.

Proof. First we check the validity of the main algorithm. As discussed in Proposition 2.4.7, we have a decomposition $fcsI = J\mathfrak{J}$, where $f \in \mathbb{Q}^\times$, $c \in \Lambda^\times$, J is a right Λ-ideal, and $\mathfrak{J}$ is a two-sided Λ-ideal. In addition, we know from the choice of $\mathcal{B}$ and the process of making sI $\mathcal{B}$-smooth, sI is integral and the norm is not divisible by the ramifying prime. Furthermore, we know that $N(fcsI) = 1$, so J is an integral right Λ-ideal, whose norm is not divisible by any ramifying prime, and $N(J) \in \mathbb{Z}^{\times 4}$.

In step (3), after the local reduction at a place $\ell \in \mathcal{D}$ and all the replacements of the ideals J and $\mathfrak{J}$, the equation $(\prod_\ell c_\ell)fcsI = J\mathfrak{J}$ remains valid, and J remains integral, while the replaced ideals J and $\mathfrak{J}$ now satisfies $v_\ell(N(J)) = v_\ell(N(\mathfrak{J})) = 0$. Therefore, at the end of step (3), after all the primes in $\mathcal{D}$ are processed, we will have $N(J) = 1$ and integral, so $J = \Lambda$. Similarly, since $N(\mathfrak{J})$ is enforced to be 1, and it is two-sided, $\mathfrak{J} = \Lambda$ as well. This means that at step (4), from the relation $(\prod_\ell c_\ell)fcsI = J\mathfrak{J} = \Lambda$, we know that $((\prod_\ell c_\ell)fcs)^{-1}$ generates I. This proves the validity of the main algorithm.

Now we verify the complexity statement. Step (1) in Algorithm 2.3.1 involves invoking the routine `NextElement` to the lattice I^{-1} repeatedly until we get a smooth element. By Heuristic 2.4.4(a), the step is probabilistic of complexity $L(\Delta)^{O(1)}$. The element $s \in I^{-1}$ as the output of this step is of size less than $N(I^{-1})L(\Delta)^{O(1)}$, by Heuristic 2.4.4(a). Therefore, sI, the input to step (2), will have an ideal norm of size $L(\Delta)^{O(1)}$. For step (2), we need to compute the G-reduction structure, which takes time $L(\Delta)^{O(1)}$ from Proposition 2.4.5. Then we need to take the `GReduce` routine, whose time complexity is agin bounded by $L(\Delta)^{O(1)}$, by Proposition 2.4.7. Note that the output ideal J involves the "extract two-sided ideal" step from sI, and we have $N(J)$ divides $N(sI)^4$, so the size of $N(J)$ is as well of the size $L(\Delta)^{O(1)}$.

Now we move to step (3), the local reduction part. If the ideal input to this step is J, then we need to local reduction at every splitting places in $\mathcal{D} = \{\ell \text{ prime} \mid \ell \text{ divides } N(J)\}$. $\mathcal{D}$ has size $\omega(N(J)) = O(\log N(J)) \log(L(\Delta)^{O(1)})$, where $\omega(N)$ denotes the number of prime divisors of N. Therefore, such number of places is at most the bit length of the norm of J, and hence it is polynomial in the bit length of Δ. And we know that the main part of the local reduction consists of Algorithms 2.5.5, 2.5.9, 2.5.14, and 2.5.18, so by combining the complexity arguments in Propositions 2.5.7 , 2.5.13, and 2.5.19, we know that the overall complexity of step 3 is polynomial in the bit length of $N(J)$, hence polynomial in the bit length of Δ. Therefore, since step (4) is negligible in time, adding all four steps in the main Algorithm 2.3.1, the total time complexity is $L(\Delta)^{O(1)}$.

Now, we provide an estimate of the size of the entries of the output generator. What we need to do is to go through again the main Algorithm 2.3.1, and sum up the precision of the elements which is applied to the output.

The first step of the main algorithm is a $\mathcal{B}$-smooth ideal sI. The way we search for s is to repeatedly invoke the routine `NextElement`(I^{-1}) to get s, until sI is $\mathcal{B}$-smooth. As in Heuristic 2.4.4, the probability each `NextElement` work is of the size $1/L(\Delta)^{O(1)}$.

After step (1) of the main algorithm, we knew that the norm of s is $N(I^{-1})L(\Delta)^{O(1)}$. Since s is chosen from the `NextElement` routine, it is reasonable to assume that the entries of s are polynomial in $N(s)$, and the bit lengths of entries of s are in $O(\ln(N(I)) + \ln(L(\Delta)))$.

In step (2), we need to extract the two-sided part. Since both $N(sI)$ and $N(J)$ are in $O(L(\Delta)^{O(1)})$, so is $f = (sI)^{-1}J$. So the bit length of f is $O(\ln(L(\Delta)))$.

Now we estimate the size of the units used in the local reduction step (3). The ideal we need for local reduction is J, and we know that $N(J)$ divides $N(sI)^4$. Therefore, if we again let $\Omega(N)$ denote the number of prime divisors of an integer N (counting multiplicity), then $\Omega(N(J)) \leq 4\Omega(N(I)) = O(\ln(L(\Delta)))$. When we make ℓ-reduction along all prime divisors of $N(J)$, it turns out that we need $\Omega(N(J))$ Bruhat-Tits traversals, which means that we need to find $\Omega(N(J))$ units as in Algorithm 2.5.9 or 2.5.14. Since 2.5.14 is the more complicated part and generates units of larger entries, we can assume all $\Omega(N(J))$ steps are from 2.5.14. Suppose in one of the steps, ℓ-adic reduction is performed, and g_i is an ℓ-adic generating global element of M_i, and $n = [\Lambda : \Lambda']$ (see Section 4.4 for the notations), Assume that $\tilde{g}_i$ is a left generator of M_i, and $\tau := \tilde{g}_i^{-1} g_i$, then since $\Lambda' \supseteq N(\tau)^4\Lambda$, we know that $n = [\Lambda : \Lambda']$ divides $N(\tau)^4$. And since we can choose $\tilde{g}_i$ by the `NextElement` routine, and from Heuristics 2.5.6, we can expect that the $1/\epsilon_2 = O(1)$-th shortest element among all possible values can be a choice of the ℓ-adic generating global element of M_i. And this implies $N(\tau) = O(1)$, as well as $n = O(1)$. And, since Algorithm 2.5.14 involves two applications of the Chinese remainder theorem, both over modulus n and ℓ, The size of the matrix g^* at step 15 will be $c\ell^2$ for some constant c. In addition, the size of entries of g_i is also polynomial in ℓ, and this implies that the entries of the output of Algorithm 2.5.14 are $O(c\ell^{\epsilon})$, for some constants c and ϵ. Multiplying all the units together, suppose that $N(J) = \prod_\ell \ell^{e_\ell}$, then the entries of the product of all the units appearing in the local reductions will have entries of size $\prod_\ell O(c^{e_\ell}\ell^{\epsilon e_\ell}) = O(c^{\Omega(N(J))}N(J)^{\epsilon}) = O(N(J)^{O(1)}) = O(L(\Delta)^{O(1)})$.

Therefore, adding the sizes of s, f, and all the units in the local step together, we know that the size, in terms of bit length, of the entries in the principal ideal generator is $O(\ln(N(I)) + \ln(L(\Delta)))$. □

2.7 Experimental results

We have implemented various steps of the algorithm in Magma. In this section, we shall demonstrate some experimental results and show how the algorithm worked out. We will first find an answer to Example 2.7.1, whose process is split into two sections: the global reduction is processed in Section 7.1; and the local reduction and the Bruhat-Tits building traversal is dealt with in Section 7.2.

Example 2.7.1. In this section and the next, we will work on the quaternion algebra $B_{p,\infty}$ over $\mathbb{Q}$ which ramifies at $p = 11$. For such a B, we can let it has a $\mathbb{Q}$-basis, $\{1, i, j, k\}$, where $i^2 = -1, j^2 = -11, ij = -ji = k$. The principal ideal problem we will consider is

$\mathrm{Mat}_2(B_{p,\infty})$. There is a maximal order $\mathcal{O}$ of $B_{p,\infty}$ with basis $\{1, i, (1+j)/2, (i+k)/2\}$. And as usual, we take $A = \mathrm{Mat}_2(B_{p,\infty})$ and $\Lambda = \mathrm{Mat}_2(\mathcal{O})$.

Now we consider the right Λ-ideal $I = 5\Lambda + \alpha\Lambda$, where

$$\alpha = \begin{bmatrix} -1 + \frac{1}{2}i + \frac{1}{2}k & -i \\ -2 - \frac{3}{2}i - \frac{1}{2}k & \frac{1}{2} - \frac{1}{2}i - \frac{1}{2}j - \frac{1}{2}k \end{bmatrix}.$$

In fact, $N(\alpha) = 10$ and $N(I) = 5$. Whenever required, assume that the set of smooth primes is $\mathcal{B} = \{2, 3, 5\}$. The goal for this section and the next is to find a right ideal generator of the princial ideal I.

2.7.1 The smoothing process and the global reduction

In this section, we will work out the global reduction process for Example 2.7.1, as described in Section 2.4. Since the ideal I is already $\mathcal{B}$-smooth, there is no need to rescale by a factor s as in step (1) of Algorithm 2.3.1. Then we move through to step (2), which involves Algorithms 2.4.2 and 2.4.6.

For Algorithm 2.4.2, we at least need to find an element c_5 in $\mathrm{Mat}_2(B)^\times$ such that $N(c_5) = 5$. As in the algorithm, we randomly generate elements in Λ with small entries (with respect to the quadratic form defined in Section 4) and collect $\mathcal{B}$-smooth elements, until c_5 can be generated by a combination. For instance, the first few short elements we generated are

$$m_1 = \begin{bmatrix} \frac{1}{2} - \frac{5}{2}i - \frac{1}{2}j + \frac{1}{2}k & \frac{1}{2} + i - \frac{1}{2}j \\ 2i & -\frac{1}{2} - \frac{1}{2}i + \frac{1}{2}j - \frac{1}{2}k \end{bmatrix}, m_2 = \begin{bmatrix} \frac{5}{2} + \frac{1}{2}i + \frac{1}{2}j + \frac{1}{2}k & -\frac{5}{2} - \frac{1}{2}i - \frac{1}{2}j - \frac{1}{2}k \\ \frac{5}{2} + \frac{1}{2}i + \frac{1}{2}j + \frac{1}{2}k & \frac{1}{2} - \frac{1}{2}i - \frac{1}{2}j - \frac{1}{2}k \end{bmatrix},$$
$$m_3 = \begin{bmatrix} -\frac{1}{2} + i + \frac{1}{2}j & 3 \\ \frac{1}{2} - \frac{3}{2}i + \frac{1}{2}j - \frac{1}{2}k & \frac{1}{2} - \frac{5}{2}i + \frac{1}{2}j + \frac{1}{2}k \end{bmatrix}, m_4 = \begin{bmatrix} i - j & 0 \\ -3i & -1 - 2i. \end{bmatrix}$$

Then the reduced norm of m_1, m_2, m_3 and m_4 are $2^2 \cdot 3^2, 2^2 \cdot 3^3, 2 \cdot 3 \cdot 5, 2^2 \cdot 3 \cdot 5$, respectively, and then one can deduce that

$$c_5 = m_1 m_2^{-1} m_3^2 m_4^{-1} = \begin{bmatrix} -\frac{203}{180} - \frac{817}{180}i - \frac{73}{90}j + \frac{61}{180}k & -\frac{301}{120} - \frac{21}{40}i - \frac{73}{40}j + \frac{67}{120}k \\ \frac{113}{72} + \frac{533}{120}i + \frac{3}{8}j - \frac{77}{360}k & \frac{83}{30} + \frac{103}{60}i + \frac{22}{15}j - \frac{49}{60}k \end{bmatrix}$$

is an element of norm 5 as required. From Algorithm 2.4.6, we need to extract the two-sided ideal $\mathfrak{J}$ of $J = c_5^{-1}I$, which is $16200^{-1}\Lambda$. From here, we can get the ideal $J = 16200c_5^{-1}I$ and send J to the local reduction, which we will demonstrate in the next section.

2.7.2 The local reduction and the Bruhat-Tits building

We will demonstrate the Bruhat-Tits building traversal in this section on the same ideal I. However, for the c_5 we have chosen, one need to do multiple steps of traversals on all the primes in $\mathcal{B}$ since 16200 has all primes in $\mathcal{B}$ as divisors. For the sake of brevity, we will demonstrate on a smaller set

$$c_5 = \begin{bmatrix} 1 & -i \\ -\frac{1}{2} - \frac{1}{2}i + \frac{1}{2}j + \frac{1}{2}k & \frac{3}{2} - \frac{1}{2}i - \frac{1}{2}j + \frac{1}{2}k \end{bmatrix}, \mathfrak{J} = 5^{-1}\Lambda, J = 5c_5^{-1}I,$$

so that we only need to do local reduction of J at the prime $\ell = 5$ since $N(J) = 5^4$.

Now we start the local reduction for the right ideal J on $\ell = 5$. First, we need to prepare the local reduction data, as in Algorithm 2.5.5. Get a right maximal ideal of norm ℓ, from there we write down a chamber as in the algorithm. Suppose the right $\mathcal{O}$-ideal of norm 5 we generated is $\mathfrak{P} = 5\mathbb{Z} + 5i\mathbb{Z} + ((7+j)/2)\mathbb{Z} + ((3i+k)/2)\mathbb{Z}$. Then after the setup as in Algorithm 2.5.5, we can construct a set of right Λ-ideals $\Lambda \supsetneq M_1 \supsetneq M_2 \supsetneq M_3 \supsetneq \ell\Lambda$, and the right ideals $M_1, M_2, M_3, \ell\Lambda$ correspond to the $\mathbb{Z}_\ell$ lattices

$$P_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 5 \end{bmatrix}, P_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}, P_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 5 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}, P_4 = \ell P_0 = \begin{bmatrix} 5 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix},$$

respectively (the column vectors form a basis). The lattices P_1, P_2, P_3 together with the standard lattice form a chamber in the Bruhat-Tits building.

A ℓ-adic global generator of the ideal J can be computed via Algorithm 2.5.16. A valid one is

$$g_I = \begin{bmatrix} \frac{9}{2} + \frac{7}{2}i - \frac{3}{2}j + \frac{1}{2}k & 4 - \frac{1}{2}i + j + \frac{1}{2}k \\ \frac{17}{2} + 3i + \frac{1}{2}j - k & -\frac{1}{2} + i + \frac{1}{2}j - k \end{bmatrix}.$$

From there, we can compute the $\mathbb{Z}_\ell$-lattice L_4 corresponding to J and a filtration $L_0 = P_0 \supsetneq L_1 \supsetneq L_2 \supsetneq L_3 \supsetneq L_4$, which are

$$L_1^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, L_2^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 5 & 0 & 0 \\ 3 & 0 & 5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, L_3^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 5 & 0 & 0 \\ 3 & 0 & 5 & 0 \\ 4 & 0 & 0 & 5 \end{bmatrix}, L_4^{(0)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 5 & 0 & 0 \\ 3 & 0 & 5 & 0 \\ 14 & 20 & 20 & 25 \end{bmatrix}.$$

The goal is to find the correct actions to match each L_i to P_i. As $N(J) = 5^4$, we need 4 steps for the Bruhat-Tits building traversal. We will give the intermediate results in each steps as below.

1. The first step. We will follow Algorithm 2.5.9. Using the correspondence from sublattices of P_0 to elements in $\mathbb{P}^3(\mathbb{F}_\ell)$ as described in Section 5.4.1, L_1 corresponds to $[2,1,0,0]^t$ and P_1 corresponds to $[0,0,4,1]^t$. The ideal of Algorithm 5.9 is to map both of them to $[0,0,0,1]^t$.

   On the side of $[2,1,0,0]^t$, we need to find two Hermitian elements in $\Lambda^\times$ and an element in $\mathrm{SL}_2(\mathbb{Z})$. It turns out that in that order, the matrices

   $$\begin{bmatrix} 1 & \frac{1+j}{2} \\ \frac{1-j}{2} & 2 \end{bmatrix} \in H, \begin{bmatrix} -3 & 2 \\ -2 & 1 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z}), \text{ and } \begin{bmatrix} 2 & \frac{-1+j}{2} \\ \frac{-1-j}{2} & 2 \end{bmatrix} \in H$$

   sends $[2,1,0,0]^t$ to $[1,3,1,1]^t$, $[3,0,0,1]^t$, and then to $[0,0,0,3]^t$.

   We can do the same on the side of $[0,0,4,1]^t$. We can see that in that order, the matrices

   $$\begin{bmatrix} 1 & i \\ -i & 0 \end{bmatrix} \in H, \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z}), \text{ and } \begin{bmatrix} 1 & \frac{i+k}{2} \\ \frac{-i-k}{2} & 2 \end{bmatrix} \in H$$

   sends $[0,0,4,1]^t$ to $[3,1,1,4]^t$, $[4,0,0,4]^t$, and then to $[0,0,0,2]^t$.

   Finally, merging the lattices in the both sides, we get an element in $\Lambda^\times$ sending $L_1^{(0)}$ to P_1, which is

   $$t_1 = \begin{bmatrix} \frac{67}{2} + 16i + \frac{1}{2}j + 4k & 22 - 5i + 13j + 9k \\ -36 + 25i - 6j - 4k & -1 + \frac{69}{2}i - 18j + \frac{13}{2}k] \end{bmatrix}.$$

   After the action of t_1, the filtration $\{L_i\}$ is now sent to

   $$L_1^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 5 \end{bmatrix}, L_2^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 5 & 0 \\ 1 & 0 & 0 & 5 \end{bmatrix}, L_3^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 5 & 0 & 0 \\ 1 & 0 & 5 & 0 \\ 1 & 0 & 0 & 5 \end{bmatrix}, L_4^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 5 & 0 & 0 \\ 1 & 0 & 5 & 0 \\ 11 & 5 & 0 & 25 \end{bmatrix}.$$

2. The second step. For this and the rest of the steps, we will follow Algorithm 2.5.14 in Section 5.4.2. We need to invoke Algorithm 2.5.16 to compute a ℓ-local global generator for the ideal M_1 first, for which we used $g_1 = \begin{bmatrix} 1 + \frac{1}{2}i + \frac{1}{2}k & i \\ \frac{7}{2} + \frac{7}{2}i + \frac{1}{2}j - \frac{1}{2}k & \frac{7}{2}i - \frac{1}{2}k \end{bmatrix}$. Note that $N(g_1) = 5 \cdot 3^2$, which implies $\Lambda' \supseteq I + 9\Lambda$. Therefore, in the Chinese Remainder Theorem step, we need to solve congruence equations with respect to 9 and $\ell = 5$.

Using the correspondence from sublattices of P_0 to elements in $\mathbb{P}^3(\mathbb{F}_\ell)$ as described in Section 5.4.2, $g_1^{-1}L_2^{(1)}$ corresponds to $[0,3,1,0]^t$ and $g_1^{-1}P_2$ corresponds to $[2,2,4,1]^t$. The idea of Algorithm 2.5.14 is to map both of them to $[0,0,0,1]^t$.

On the side of $[0,3,1,0]^t$, since $0 \cdot 0 \neq 3 \cdot 1$, we need to find an element in $\mathrm{SL}_2(\mathbb{Z})$ and a Hermitian elements in $\Lambda^\times$ and an element in $\mathrm{SL}_2(\mathbb{Z})$. It turns out that in that order, the matrices

$$\begin{bmatrix} 10 & -189 \\ 9 & -170 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z}), \text{ and } \begin{bmatrix} 1 & -\frac{9i+9k}{2} \\ \frac{9i+9k}{2} & 244 \end{bmatrix} \in H$$

sends $[0,3,1,0]^t$ to $[1,0,0,2]^t$, and then to $[0,0,0,3]^t$.

On the side of $[2,2,4,1]^t$, we also need to find an element in $\mathrm{SL}_2(\mathbb{Z})$ and a Hermitian elements in $\Lambda^\times$ and an element in $\mathrm{SL}_2(\mathbb{Z})$. The matrices

$$\begin{bmatrix} 163 & -9 \\ -18 & 1 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z}), \text{ and } \begin{bmatrix} 244 & \frac{-9+9j}{2} \\ \frac{-9-9j}{2} & 1 \end{bmatrix} \in H$$

sends $[2,2,4,1]^t$ to $[4,0,0,1]^t$, and then to $[0,0,0,1]^t$.

Merging the lattices in the both sides, we get an element $t_2 \in \Lambda_1^\times$ sending $L_2^{(1)}$ to P_2, which is

$$\begin{bmatrix} -\frac{534416557}{2} + \frac{880668981}{2}i + \frac{583250271}{2}j - \frac{559225287}{2}k & -\frac{850696553}{2} + \frac{124684578}{5}i - \frac{19854211}{2}j + \frac{430688781}{5}k \\ -\frac{5738659839}{2} - 337452543i + \frac{1662152013}{2}j - 1220649876k & -1381203482 + \frac{1317794625}{2}i - 291087732j + \frac{563140125}{2}k \end{bmatrix}.$$

After the action of t_2, the filtration $\{L_i\}$ is now sent to

$$L_1^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 5 \end{bmatrix}, L_2^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}, L_3^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 5 & 10 & 5 & 25 \end{bmatrix}, L_4^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 30 & 85 & 55 & 125 \end{bmatrix}.$$

3. The third step. Again we will follow Algorithm 2.5.14 in Section 5.4.2. Slightly different from the second step, the right-ideal generator of M_2 is known because of our selected form, which is $g_2 = \begin{bmatrix} 1 & 0 \\ 0 & 5 \end{bmatrix}$. And since it is a generator, $\Lambda' = \Lambda$. Therefore, we do not need the Chinese Remainder Theorem for the pullback in Algorithm 2.5.14.

   Using the correspondence from sublattices of P_0 to elements in $\mathbb{P}^3(\mathbb{F}_\ell)$ as described in Section 5.4.2, $g_2^{-1}L_3^{(2)}$ corresponds to $[4,3,4,1]^t$ and $g_2^{-1}P_3$ corresponds to $[4,1,0,0]^t$. We need to map both of them to $[0,0,0,1]^t$.

On the side of $[4,3,4,1]^t$, we get the matrices $\begin{bmatrix} -3 & 2 \\ -2 & 1 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z})$ and $\begin{bmatrix} 4 & \frac{i+k}{2} \\ \frac{-i-k}{2} & 1 \end{bmatrix} \in H$ sends $[4,3,4,1]^t$ to $[2,0,0,1]^t$, and then to $[0,0,0,1]^t$. On the side of $[4,1,0,0]^t$, $\begin{bmatrix} 1 & i \\ -i & 2 \end{bmatrix} \in H$, $\begin{bmatrix} -1 & -1 \\ 2 & 1 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z})$, and $\begin{bmatrix} 4 & \frac{i+k}{2} \\ \frac{-i-k}{2} & 1 \end{bmatrix} \in H$ sends $[4,1,0,0]^t$ to $[3,2,3,1]^t$, then to $[2,0,0,1]^t$, and then to $[0,0,0,1]^t$.

Merging the lattices in the both sides, we get $t_3 = \begin{bmatrix} -10-8i & \frac{6}{5}+i \\ 40-25i & -5+3i \end{bmatrix} \in \Lambda_2^\times$ sending $L_3^{(2)}$ to P_3. After the action of t_3, the filtration $\{L_i\}$ is now sent to

$$L_1^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 5 \end{bmatrix}, L_2^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}, L_3^{(3)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 5 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}, L_4^{(3)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 10 & 10 & 25 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}.$$

4. The fourth (and final) step. Similar to the second step, we will need to use Algorithm 2.5.16 to find an ℓ-adic global generator for M_3. A candidate is $g_3 = \begin{bmatrix} \frac{7}{2}+\frac{1}{2}j & \frac{3}{2}+\frac{3}{2}i-\frac{1}{2}j+\frac{1}{2}k \\ 5 & \frac{5}{2}+\frac{5}{2}i-\frac{5}{2}j+\frac{5}{2}k \end{bmatrix}$. As $N(g_3) = 5^3 \cdot 2^3$, and in the Chinese Remainder Theorem step in Algorithm 2.5.14, we need to solve congruence equations with respect to 8 and $\ell = 5$.

   Using the correspondence from sublattices of P_0 to elements in $\mathbb{P}^3(\mathbb{F}_\ell)$ as described in Section 5.4.2, $g_2^{-1}L_3^{(2)}$ corresponds to $[3,2,1,1]^t$ and $g_2^{-1}P_3$ corresponds to $[1,2,1,1]^t$. We need to map both of them to $[0,0,0,1]^t$.

   For $[3,2,1,1]^t$, the matrices $\begin{bmatrix} -127 & 16 \\ -8 & 1 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z})$ and $\begin{bmatrix} 769 & -8+8j \\ -8-8j & 1 \end{bmatrix} \in H$ sends $[3,2,1,1]^t$ to $[1,0,0,1]^t$, and then to $[0,0,0,1]^t$. On the side of $[1,2,1,1]^t$, $\begin{bmatrix} 129 & -16 \\ -8 & 1 \end{bmatrix} \in \mathrm{SL}_2(\mathbb{Z})$ and $\begin{bmatrix} 769 & 8-8j \\ 8+8j & 1 \end{bmatrix} \in H$ sends $[1,2,1,1]^t$ to $[4,0,0,1]^t$ and then to $[0,0,0,1]^t$.

   Merging the lattices in the both sides, we get $t_4 \in \Lambda_3^\times$, which is

$$\begin{bmatrix} 649411557 - \frac{539320764}{5}i + 39070140j + \frac{904692092}{5}k & -\frac{703119492}{5} + \frac{561665612}{5}i - \frac{158492436}{5}j - \frac{177774436}{5}k \\ 2270519176 - 1361698372i - 266298448j + 577765196k & -649802211 + 579280672i - 14068540j - 68263256k \end{bmatrix},$$

   and sends $L_4^{(3)}$ to $5P_0$, homothetic to P_0.

As the Bruhat-Tits building traversal is completed, we have the relation

$$t_4 t_3 t_2 t_1 (5c_5^{-1} I) = 5\Lambda,$$

and can conclude that $c_5(t_4t_3t_2t_1)^{-1} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ is a generator of I, where

$$\begin{bmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \end{bmatrix} = \begin{bmatrix} \frac{15048643669663719 88561}{2} + 2616398492156972754228i + \frac{6883836707729554704 27}{2}j - 1177005827499222504624k \\ -6590989077438772424 - \frac{21711336825261827068 77}{2}i - 1338855338573649046 64j + \frac{33992513987743327747 5}{2}k \\ 611100321503819701318 - \frac{24947374597414364195167}{2}i + 1778592087129882929431j - \frac{15513734241466967765 49}{2}k \\ \frac{16096721611166427568 91}{2} + 2646154544490715747206i - \frac{79012245230745201685 7}{2}j + 6457930797167690550 10k \end{bmatrix}.$$

2.8 Future directions

How far can we generate the main algorithm? It is natural to ask if we can solve the principal ideal problem for $\mathrm{Mat}_n(B)$, where B is a quaternion algebra over a number field K. The strong approximation and Page's algorithm in [Pag14] suggests us to consider the norm ideal $N(I) \subseteq \mathcal{O}_K$. We did not generate to that level since $\mathrm{Mat}_2(B_{p,\infty})$ is the setting which provides us some link to applications.

In the local reduction in Section 2.5, we excluded the ramified case $\ell = p$. Can we do local reduction on a ramified prime?

Chapter 3
Computing Igusa Polynomials via p-Adic Methods

3.1 Introduction

3.1.1 The case of genus 1: CM elliptic curves and Hilbert class polynomials

The key step to construct the system parameter for elliptic curve cryptography is to construct an elliptic curve E over a finite field $\mathbb{F}_q$. To ensure the hardness of the underlying discrete logarithm problem, we need the group $E(\mathbb{F}_q)$ to contain a subgroup of a large prime order (ideally, we want $|E(\mathbb{F}_q)|$ to be a prime).

The standard approach is to pick a random curve E until we find one with $|E(\mathbb{F}_q)|$ satisfying our condition. Alternatively, one can achieve this via the complex multiplication (CM) method.

We give a quick overview of how the CM method works in finding the system parameter. Suppose we start with a curve $\tilde{E}$ over a number field L, such that $\mathrm{End}_{\overline{\mathbb{Q}}}(E) \cong \mathcal{O}_K$, where $\mathcal{O}_K$ us the maximal order of an imaginary quadratic field K. In this case, we say that $\tilde{E}$ has CM by K. Let $\mathfrak{p}$ be a prime ideal of L with absolute norm p^f. It is a well-known result (see, for instance, [Lan87, Chapter 13, Theorem 12]) that $\tilde{E}$ has ordinary reduction if and only if p splits in the CM field K. Now, suppose that $\tilde{E}$ has ordinary reduction to E over the finite field $\mathbb{F}_q = \mathbb{F}_{p^f}$. Let π be an element in $\mathcal{O}_K$ such that $\pi\bar{\pi} = p^f$. Then the Weil-p^f number of E is either π or $-\pi$, which implies $|E(\mathbb{F}_{p^f})| = p^f + 1 \pm \mathrm{Tr}_{K/\mathbb{Q}}(\pi)$.

According to the construction above, finding a curve over a finite field with prescribed order just boils down to choosing the parameters K, p, f, an elliptic curve over a number field with CM by K, and then taking reduction.

For a fixed imaginary quadratic field K, the Hilbert class polynomial encodes all elliptic curves over number fields with CM by K. For $j \in \overline{\mathbb{Q}}$, denote E_j by an elliptic curve over $\overline{\mathbb{Q}}$ whose j-invariant is j. Then the Hilbert class polynomial of K is defined as

$$H_K(x) = \prod_{\substack{j \\ \mathrm{End}(E_j)=\mathcal{O}_K}} (x - j).$$

It is a direct consequence of the previous discussion that an application of Hilbert class polynomial is the construction of system parameters in the elliptic curve cryptography. It also has importance on the theoretical side: Indeed, $H_K \in \mathbb{Z}[x]$, and the splitting field of H_K in K is the Hilbert class field of K.

It is then natural to ask for the generalization for higher dimension abelian varieties. In this chapter, our main focus will be on dimension $g = 2$. To construct the analog of Hilbert class polynomials, we need the analogy of j-invariant which characterizes abelian varieties of genus 2.

3.1.2 The case of genus 2: CM hyperelliptic Jacobians and Igusa class polynomials

A CM field is a totally imaginary extension of a totally positive number field. For a quartic CM field K, we say that an abelian surface A defined over a number field L has CM by K if $\mathrm{End}_{\overline{L}}(A) \cong \mathcal{O}_K$. Here we want to construct a "class polynomial" whose roots are some invariants that characterize abelian surfaces with CM by K.

Instead of a general abelian variety, we are more interested when the abelian variety is isomorphic to the Jacobian variety of a hyperelliptic curve, since more operations, such as the point addition and some isogenies can be computed explicitly in this case.

It turned out that the situation is easier in lower genus, since every principally polarized abelian variety of dimension $g \leq 3$ is isomorphic to the Jacobian variety of an algebraic curve ([OU73]), and in particular, every simple abelian surface is isomorphic to the Jacobian of a hyperelliptic curve of genus 2. If we further restrict the quartic CM field K to be primitive (here it simply means non-biquadratic), then every curve with CM by K will be simple, and hence can be represented as a hyperelliptic Jacobian.

Therefore, it suffices to find all elliptic curves of genus 2 whose Jacobian variety has CM by $\mathcal{O}_K$. Since the coarse moduli space of genus g curves has dimension $3g - 3$, we would expect 3 parameters to characterize a hyperelliptic curve. There are several such choices, but we will use the Igusa invariants in [Igu60]. Conversely, given the Igusa invariants (i_1, i_2, i_3), given that the Jacobian variety is simple, we can use Mestre's algorithm in [Mes91] to recover the hyperelliptic curve.

Using the Igusa invariants (i_1, i_2, i_3) described above, for each CM quartic field K, one can define three polynomials $H_{K,1}, H_{K,2}, H_{K,3}$, each of them is a monic polynomial whose roots are exactly the corresponding Igusa invariant. We call these the Igusa polynomials.

Not every property from the Hilbert class polynomials generalizes to the Igusa class polynomials. First, the Igusa polynomials are no longer of integral coefficients. They are still of rational coefficients, and the prime divisors of the denominators are those primes that have split Jacobians after reduction (see [Str10, Theorem 10.1]). The Igusa polynomials no longer give information on the Hilbert class field of K; instead, from K, we can compute a reflex field of K. And it turns out that the reflex field adjoining the Igusa invariants is a subfield of the Hilbert class field of the reflex field. For more details on the properties, see [Shi98] or [Spa94].

Analogous to the genus 1 case, one construct cryptographically secure hyperelliptic curves over finite fields via Igusa polynomials. Suppose that a quartic CM field and p, f are chosen such that there exists a hyperelliptic Jacobian over a number field which has CM by K and an ordinary reduction over the finite field $\mathbb{F}_{p^f}$. Then the order of the reduced hyperelliptic Jacobian will be $N_{K/\mathbb{Q}}(\pi - 1)$ for some $\pi \in \mathcal{O}_K$ satisfying $\pi\overline{\pi} = p^f$.

We will now discuss some approaches in computing the Igusa class polynomials, but first, we shall look at the case of Hilbert class polynomials in genus 1.

Hilbert class polynomials in genus 1 and the *p*-adic method. Algorithms in computing Hilbert class polynomials for imaginary quadratic fields K can be classified in the following three categories:

1. The analytic approach. The idea is to find a set of representatives $\{\tau_{\mathfrak{a}}\}_{\mathfrak{a}\in \mathrm{Cl}(K)}$ on the upper half-plane, compute the j-invariants $j(\tau_{\mathfrak{a}})$, and multiply the factors $(x - j(\tau_{\mathfrak{a}}))$ to recover the Hilbert class polynomial H_K. See [Eng09] for a discussion of possible improvements and the complexity arguments.

2. The CRT approach (see [ALV04]). The idea is to start by determining a set of rational primes S_K, such that for each prime $p \in S_K$, there are $h(K)$ isomorphism classes of an elliptic curve over $\mathbb{F}_p$ with CM by K. Suppose the j-invariants are $\{j_{p,i}\}_{1\leq i \leq \mathrm{Cl}(K)}$. Then $\prod(x - j_{p,i})$ will coincide with $H_K \pmod{p}$. Given that the set S_K is sufficiently large, one can recover the Hilbert class polynomial by the Chinese Remainder Theorem.

3. The p-adic approach.

As our goal is to work on the p-adic method for higher genus, we summarize the p-adic method in genus 1 in greater detail. This approach is described and analyzed by Bröker in [Brö08].

To compute the Hilbert class polynomial for an imaginary quartic field K, We start by choosing a prime p that splits completely in H, the Hilbert class field of K. Since such a p also splits in K, Deuring's theorem implies that elliptic curves with CM by $\mathcal{O}_K$ remain ordinary upon reduction by p. Conversely, for such a p, one can find an elliptic curve over $\mathbb{F}_p$ with endomorphism ring $\mathcal{O}_K$.

After E_p is found, we need to find the "canonical lifting", denoted by $\tilde{E}$, of E_p. Denote by $\mathrm{Ell}_K(\mathbb{F}_p)$ and $\mathrm{Ell}_K(\mathbb{Q}_p)$ the sets of j-invariants of elliptic curves with endomorphism rings $\mathcal{O}_K$ over $\mathbb{F}_p$ and $\mathbb{Q}_p$, respectively. It is known that $\mathrm{Ell}_K(\mathbb{F}_p)$ and $\mathrm{Ell}_K(\mathbb{Q}_p)$ are bijective, which means that each $\tilde{E} \in \mathrm{Ell}_K(\mathbb{Q}_p)$ reduces to a $E \in \mathrm{Ell}_K(\mathbb{F}_p)$, and it turns out that $\tilde{E}$ the canonical lifting of E. In [Brö08], Bröker considered the analytic space $X_K(\mathbb{C}_p) := \{j \in \mathbb{C}_p \mid \mathrm{red}(j) \in \mathrm{Ell}_K(\mathbb{F}_p)\}$. Given an ideal $I \subset \mathcal{O}_K$, there is an action on the CM curves, $\rho_I : \mathrm{Ell}_K(\mathbb{Q}_p) \to \mathrm{Ell}_K(\mathbb{Q}_p)$. Bröker claimed that the map can be extend to an analytic map $\rho_I : X_K(\mathbb{C}_p) \to X_K(\mathbb{C}_p)$, whose fixed points are exactly the j-invariants of the canonical liftings, and can be computed by Newton's method. And after one canonical lifting of CM curve is computed, the rest of them can be computed via the action of the class group $\mathrm{Cl}(\mathcal{O}_K)$.

Igusa class polynomials in genus 2 and the p-adic method.

All the three approaches in the genus 1 case had been tried out in genus 2. They are:

1. The analytic approach. The first analytic algorithm with a complete analysis of complexity can be dated back to Streng in [Str14]. For a given quartic CM field K, the idea is to first give a list of complex abelian surfaces with CM by K. They will be given by the complex lattice $A_{\mathfrak{a}} = \mathbb{C}^2/\Phi(\mathfrak{a})$, where $\mathfrak{a}$ is an ideal in $\mathcal{O}_K$ and Φ is a CM type. Via the analytic theta functions, one can recover the Igusa invariants. And after compute the Igusa invariant through all the isomorphism classes, one can recover the Igusa class polynomial via the LLL algorithm. Improvements are proposed by Enge and Thomé in [ET14].

2. The CRT approach, first proposed by Eisentraeger and Lauter in [EL10]. The idea is similar: Find primes p such that there are sufficiently many hyperelliptic Jacobians defined over $\mathbb{F}_p$, and find all such hyperelliptic curves, reconstruct $H_{K,i} \pmod{p}$, and use CRT to recover the Igusa polynomials. See also [BGL11] for some suggested improvements using correspondences on Siegel modular varieties.

3. The p-adic method.

Our main goal in Chapter 4 is to investigate the possibilities of the p-adic approach. The standard approach of using the p-adic method to compute the Igusa class polynomial of a

CM field K proceeds as follows: (1) Search for an ordinary abelian surface A over a finite field $\mathbb{F}_{p^r}$ so that A has CM by K; (2) Find the canonical lifting $\tilde{A}$ of A. More explicitly, we are in search for an abelian surface $\tilde{A}$ over d $\mathbb{Q}_{p^{r'}}$ such that $\mathrm{End}(\tilde{A}) = \mathrm{End}(A)$, where the field of definition of $\tilde{A}$ the degree r' unramified extension of $\mathbb{Q}_p$, such that $\mathrm{End}(\tilde{A}) \cong \mathcal{O}_K$; (3) Recover the Igusa invariants and Igusa polynomials from $\tilde{A}$.

It turns out that the critical step of the p-adic approach is step (2) above, for which we will take $\tilde{A}$ as the canonical lift of A (see Definition 3.2.10 for details). Since the canonical lift also lifts the p-power Frobenius to itself, this leads to the condition that the p-power Frobenius have to lift to the canonical lift $\tilde{A}$, which will be a (p, p)-isogeny. This is the main obstruction to generalize the p-adic method to genus 2 since it is in general hard to describe the correspondence of (p, p)-isogenies on the moduli space of higher-dimensional abelian varieties.

In [GHK$^+$06], Gaudry et al. gave an approach for $p = 2$ on Rosenhein invariants. For $p = 3$, [CKL08] proposed a method using the 4-theta null points, but their method does not generate. For a general method, [CL09] proposed a method for general p and any dimension g using $2^n p$-theta null points, but it turned out to be feasible in practice only when $p = 3$ since computing $2^n p$-theta null points involves the invocation of computationally heavy Gröbner basis algorithms. Faugére et al. discussed improvements of the Gröbner basis step in [DJP14] and gave experimental results to compute $2p$-theta null points for $p = 3$ or 5.

3.1.3 Outline

Our main contribution to this topic is to follow the algorithmic structure in Section 3.3, we provided improvements on various steps. While it was only practical in the past literature, such as [GHK$^+$06] and [CKL08] to apply the p-adic method to compute the Igusa class polynomial when $p = 2, 3$, we were able to run examples, under our improvements, up to $p = 5$. The central ingredient of the p-adic method is the canonical lifting of a hyperelliptic Jacobian over a finite field to a hyperelliptic Jacobian over an unramified extension of a p-adic field. Our main contribution is the improvements that support Theorem 3.1.1 for canonical lifts and Theorem 3.1.2 for the entire Igusa polynomial algorithm.

Theorem 3.1.1. Let K be a quartic CM field and K_0 be the real subfield of K. Suppose that the discriminant of K and K_0 are $D_0^2 D_1$ and D_0, respectively. Suppose a hyperelliptic curve C of genus 2, defined over some finite field $\mathbb{F}_q = \mathbb{F}_{p^r}$ is given, such that the endomorphism ring of $\mathrm{Jac}(C)$ is the maximal order $\mathcal{O}_K$ of K.

Assume that the $2p$-theta null point of $\mathrm{Jac}(C)$ is in the finite field extension $\mathbb{F}_{q^d}$. If ω is the exponential factor, such that multiplying two $m \times m$ matrices takes time $O(m^\omega)$, and μ

is the exponential factor such that multiplying two m'-bit integers is $O(m'^{\mu})$. The canonical lift of $\mathrm{Jac}(C)$ can be computed in

$$\Xi = O\left(p^{8(\omega((p+1)+1))}\right) + O\left(d^{\mu}(p-2)!\left(\frac{p-1}{2}\right)^{p}\right) + \tilde{O}\left((D_0^{5/2}D_1^{3/2}d)^{\mu}\right)$$

operations on the finite field $\mathbb{F}_q$. The space complexity of the algorithm is $O(p^8)$ elements in the finite field $\mathbb{F}_q$.

Theorem 3.1.2. Let K and K_0, as well as all other notations, be as in Theorem 3.1.1. Suppose in addition that X is the time complexity to verify whether an abelian variety over a finite field has CM by K (we have $X = O(q^{18})$ if the algorithm proposed by Freeman and Lauter in [FL08] is used). Then the Igusa class polynomials $(h_{K,1}, h_{K,2}, h_{K,3})$ can be computed in time and space complexity

$$O\left(\frac{q^3(\log q)^9}{\sqrt{D_0 D_1}}\right) + O\left(\frac{q^{3/2}X}{\sqrt{D_0 D_1}}\right) + \Xi + \tilde{O}(D_0^7 D_1^5).$$

The remaining of the chapter will be organized as follows. In Section 3.2, we will introduce the background related to CM abelian varieties and algebraic theta functions. We will propose the main algorithm in Section 3.3, which inputs a CM quartic field and outputs the Igusa class polynomial, and give a brief overview of the major steps. The major steps, finding curves with the correct CM over a finite field, finding the canonical lift, and finishing with the Igusa class polynomial, will be elaborated in Sections 3.4, 3.5, and 3.6, respectively. We will carry out actual examples in section 3.7. The validity, computational complexity, and other issues regarding the main algorithm will be discussed in section 3.8.

3.2 Background

3.2.1 Moduli of Abelian Surfaces and Moduli of Hyperelliptic Curves

3.2.1.1 Principally Polarized Abelian and Jacobian Varieties; The Moduli Problem

The case of moduli of Jacobian of hyperelliptic curves is described as a special case in [CO12, Chapter 2].

Let k be an algebraically closed field. Denote by $\mathscr{A}_{g,1}$ be the moduli space of principally polarized abelian varieties (ppav) of dimension g over k.

The main object of our interest in $\mathscr{A}_{g,1}$ are those which are Jacobians. We say $\mathcal{C}/k$ is a curve of compact type if it satisfies two conditions: (i) every irreducible component is smooth, and (ii) The dual graph of $\mathcal{C}$ is a tree. A curve of compact type gives a compact Jacobian variety $\mathcal{J} = \mathrm{Jac}(\mathcal{C})$ which is a dimension g abelian variety (loc. cit.).

$\mathcal{J}$ admits a principal polarization λ whose construction is described, for instance, in Milne's notes, [Mil08, sections 1.8, 3.6]. First fix a point $P \in \mathcal{C}(k)$. Then there is a morphism $f^r : \mathcal{C}^r \to \mathcal{J}$ which maps $(P_1, \cdots, P_r) \mapsto [P_1 + \cdots + P_r - r \cdot P]$, and the morphism f^r induces a map $f^{(r)} : \mathcal{C}^{(r)} \to \mathcal{J}$, where $\mathcal{C}^{(r)}$ is the r-th symmetric power of $\mathcal{C}$. Then $W^r := \mathrm{Im}(f^{(r)})$ is a subvariety of $\mathcal{C}^{(r)}$, and we define the theta divisor $\Theta := W^{g-1}$ (this is defined up to a translation as we vary the initial selection of P). It is shown in [loc. cit., Theorem 3.6.6] that the invertible sheaf $\mathcal{L}(\Theta)$ defines a principal polarization:

$$\begin{aligned} \lambda = \lambda : \mathcal{J} \to \mathcal{J}^\vee \cong \quad & \mathrm{Pic}^0(\mathcal{J}) \\ a \mapsto \quad & t_a^* \mathcal{L}(\Theta) \otimes \mathcal{L}(\Theta)^{-1}. \end{aligned} \tag{3.1}$$

Here $\mathrm{Pic}^0(A) \subset \mathrm{Pic}(A)$ is the collection of invertible line bundles $\mathcal{L}$ on A such that $t_a \mathcal{L} \otimes \mathcal{L}^{-1} = 0 \in \mathrm{Pic}(A)$ for all $a \in A$.

We denote by $\mathscr{M}_g$ the moduli space of genus g curves of compact type. From [CO12, Chapter 2], there is a morphism $j : \mathscr{M}_g \to \mathscr{A}_{g,1}$ that sends $\mathcal{C}$ to $j(\mathcal{C}) = (\mathrm{Jac}(\mathcal{C}), \lambda_{\mathcal{C}})$. It is known that while the genus $g \geq 2$, the dimension of $\mathscr{M}_g$ is $3g - 3$ while the genus $\mathscr{A}_{g,1}$ is $g(g+1)/2$, so at genus 4 or larger, it is not reasonable to expect that every ppav comes from a Jacobian. But at low genera we have:

Theorem 3.2.1. [Oort, Ueno [OU73]] Suppose $g \leq 3$, then $j(\mathscr{M}_g) = \mathscr{A}_{g,1}$. In other words, every ppav of dimension 3 or less is isomorphic to a Jacobian variety of a (not necessarily irreducible) curve of genus g. (See also [CO12, Chapter 2]).

3.2.1.2 The Igusa Invariants

The invariants are described by Igusa [Igu60]. For more details, see [Str14, Chapter 2] or [BGL11, Chapter 2].

Let k be a field whose characteristic is not 2. Suppose we start with a hyperelliptic curve over k of genus 2, defined by the equation $y^2 = c(x - a_1)(x - a_2) \cdots (x - a_6)$, and $a_1, \cdots a_6 \in \bar{k}$ are pairwise distinct.

Let us treat a_i as an indeterminate for now. For $1 \leq i, j \leq 6$, denote for brevity $(ij) = a_i - a_j \in \mathbb{Z}[a_1, \cdots, a_6]$. There are homogeneous and symmetric Igusa-Clebsh invariants of degree 2, 4, 6, and 10 described as below:

$$\begin{aligned}
I_2 &:= c^2 \sum_{O_{15}} (12)^2(34)^2(56)^2; \\
I_4 &:= c^4 \sum_{O_{10}} (12)^2(23)^2(31)^2(45)^2(56)^2(64)^2; \\
I_6 &:= c^6 \sum_{O_{60}} (12)^2(23)^2(31)^2(45)^2(56)^2(64)^2(14)^2(25)^2(36)^2; \\
I_{10} &:= c^{10} \prod_{i<j} (ij)^2.
\end{aligned} \tag{3.2}$$

Note that the summation in (3.2) ranges over the orbits of the action of the symmetric group S_6 on the summands in $\mathbb{Z}[a_1, \cdots, a_6]$, so I_2, I_4 and I_6 is a summation of 15, 10, and 60 terms, respectively. Also note that $I_2, I_4, I_6, I_{10} \in k$ because of the symmetry in the summation. Then the module space $\mathscr{M}_2$ has the following structure:

Theorem 3.2.2 (Igusa [Igu60]). $\mathscr{M}_2$ is isomorphic to $\mathbb{P}_{2,4,6,10} \backslash H_{10}$, where $\mathbb{P}_{2,4,6,10}$ is the 3-dimensional $2, 4, 6, 10$-weighted projective plane, and H_{10} is the weighted hyperplane generated by the weight 10 variable.

The isomorphism is given by sending a hyperelliptic curve to its Igusa-Clebsh invariants $[I_2 : I_4 : I_6 : I_{10}]$.

When the characteristic of k is not 2, the absolute Igusa invariants (i_1, i_2, i_3) is obtained by a regular map from $\mathbb{P}_{2,4,6,10} \backslash H_{10}$ to the affine space $\mathbb{A}_k^3$. Different maps are being chosen in different literature: for instance, Bröker et al. used $(i_1, i_2, i_3) := (I_2^5/I_{10}, I_4 I_2^3/I_{10}, I_6 I_2^2/I_{10})$ in [BGL11]; while Streng in [Str14] suggested $(I_4 I_6'/I_{10}, I_2 I_4^2/I_{10}, I_4^5/I_{10}^2)$, where $I_6' := (I_2 I_4 - 3I_6)/2$.

In our work, we will follow Kohel's suggestions in the Echidna library in [Koh], which takes $(i_1, i_2, i_3) = (I_4 I_6/I_{10}, I_2^3 I_4/I_{10}, I_2^2 I_6/I_{10})$.

3.2.1.3 The Igusa Class Polynomial

Analogous to the Hilbert class polynomial in the case of genus 1, for the case of genus 2, we can construct a triple of Igusa class polynomial as follows:

Definition 3.2.3 (Igusa class polynomial)**.**

$$h_{K,n}(x) := \prod_{\mathcal{C} \in S_{\mathcal{O}_K}} (x - i_n(\mathcal{C})), \quad n \in \{1, 2, 3\}, \tag{3.3}$$

where $S_{\mathcal{O}_K}$ means the isomorphism classes of hyperelliptic curves over $\mathcal{O}_K$ whose Jacobians have CM by $\mathcal{O}_K$.

One thing to note is that unlike in the genus 1 case, the Igusa class invariants need not be an algebraic integer, hence $h_{K,n}(x)$ is only guaranteed to be in $\mathbb{Q}[x]$.

There is another issue with the Igusa class polynomial. Since we are not "labeling" the Igusa class invariants, if we denote $h_K^* = |S_{\mathcal{O}_K}|$, the roots of the Igusa class polynomials suggests h_K^{*3} triples (i_1, i_2, i_3), but only h_K^* of them gives a curve with CM by $\mathcal{O}_K$, and we have no more efficient way to figure them out other than exhaustive search. An improvement for this issue is suggested in [GHK$^+$06, Chapter 3] and [Str14, section 2.4]. Here we let $\mathcal{C}_1, \cdots, \mathcal{C}_{h_K^*}$ be a set of representatives in $|S_{\mathcal{O}_K}|$.

The main ingredient of the improvement is that we need to replace the polynomials $h_{K,2}$ and $h_{K,3}$ by some function $\tilde{h}_{K,2}$ and $\tilde{h}_{K,3}$ such that $\tilde{h}_{K,n}(i_1(\mathcal{C}_j)) = i_n(\mathcal{C}_j)$ for $n \in \{2,3\}$. This can be computed by Lagrange interpolation in the case that we have all $i_1(\mathcal{C}_j)$ distinct (which happens most of the cases):

$$\tilde{h}_{K,n}(x) = \sum_{j=1}^{h_K^*} \left(i_n(\mathcal{C}_j) \prod_{\substack{l=1\\ l\neq j}}^{h_K^*} \frac{x - i_1(\mathcal{C}_l)}{i_1(\mathcal{C}_j) - i_1(\mathcal{C}_l)} \right), \quad n \in \{2,3\}.$$

Observe that $h'_{K,1}(i_1(\mathcal{C}_j)) = \prod_{l=1, l\neq j}^{h_K^*} (i_1(\mathcal{C}_j) - i_1(\mathcal{C}_l))$. This means that we can replace $\tilde{h}_{K,n}(x)$ with functions $\hat{h}_{K,n}(x)$, $n \in \{2,3\}$ with potentially smaller coefficients:

$$\hat{h}_{K,n}(x) = \sum_{j=1}^{h_K^*} \left(i_n(\mathcal{C}_j) \prod_{\substack{l=1\\ l\neq j}}^{h_K^*} (x - i_1(\mathcal{C}_l)) \right), \quad n \in \{2,3\}. \tag{3.4}$$

From the triple of polynomials $(h_{K,1}, \hat{h}_{K,2}, \hat{h}_{K,3})$, given $i_1(\mathcal{C}_j)$ for some j, we can obtain $i_n(\mathcal{C}_j)$ by $i_n(\mathcal{C}_j) = \hat{h}_{K,n}(i_1(\mathcal{C}_j))/h'_{K,1}(i_1(\mathcal{C}_j))$.

3.2.2 The Theory of CM

The object we concern about is the CM abelian surfaces. To give a better picture of how things are generalized, we briefly discuss the case of dimension $g = 1$. A standard reference for CM elliptic curves is [Sil94]; for general CM abelian varieties, see [Shi98].

3.2.2.1 CM elliptic curves

Suppose $K = \mathbb{Q}(\sqrt{D})$ is an imaginary quadratic field of discriminant $D < 0$, and $\mathcal{O}_K$ be its maximal order. We say that an abelian variety A defined over L has CM by K if $\mathrm{End}_{\overline{L}}(A) = \mathcal{O}_K$.

Now we return to the elliptic curve case. Suppose E is defined over $\mathbb{C}$ and has CM by K. Then one have $E(\mathbb{C}) \cong \mathbb{C}/\mathfrak{a}$ for some integral ideal $\mathfrak{a} \subseteq K$. The main theorem of CM theory implies that E can be defined over H, the Hilber class field of K. In addition, there is a bijection

$$\left\{\begin{matrix}\text{Isomorphism classes of} \\ \text{CM elliptic curves}\end{matrix}\right\} \leftrightarrow \mathrm{Cl}(K),$$

where both $\mathrm{Gal}(H/K)$ and $\mathrm{Cl}(K)$ act transitively on each of the two sets. And it implies that if we define the Hilbert class polynomial as the monic polynomial H_K whose roots are exactly the j-invariants of the curves E with CM by K, then the splitting field of H_K is the Hilbert class field.

For our application, we also need to know about the reduction of CM elliptic curves over $\overline{\mathbb{Q}}$. Suppose K and H are as above, and E is an elliptic curve with CM by K, defined over H. Suppose $\mathfrak{p}$ is prime ideal in H and $\mathfrak{p} \cap \mathbb{Z} = (p)$. Assume that E has a good reduction at $\mathfrak{p}$. Then from [Lan87, Chapter 13, Theorem 12], E has an ordinary reduction if p splits in K; and has supersingular reduction if p ramifies or is inert in K.

3.2.2.2 CM abelian varieties

For the case of dimension $g > 1$, we first need to define general CM fields and CM types.

Definition 3.2.4 (CM fields and CM types)**.** Let K be a number field and $[K : \mathbb{Q}] = 2g$. We say that K is a CM field if K is a totally imaginary extension of a totally real number field K_0 of degree g.

A CM type of K is a collection of g embeddings: $\Phi = \{\phi_1, \cdots, \phi_g\}$, $\phi_i : K \hookrightarrow \mathbb{C}$, such that for every embedding $\psi : K \hookrightarrow$, exactly one of ψ or $\bar{\psi}$ (the complex conjugate of ψ) is contained in Φ. Φ can be viewed as a map $K \to \mathbb{C}^g$.

Suppose $K' \subsetneq K$ is a CM field and Φ' is a CM type of K'. We say that (K, Φ) is induced from (K', Φ'), if $\{\phi'\}_{\phi' \in \Phi'} = \{\phi|_{K'}\}_{\phi \in \Phi}$. We call (K, Φ) a primitive CM type if it is not induced from a CM subfield.

For a CM type (K, Φ), the reflex field of K, denoted as $K^\dagger$, is a CM field which is the fixed field of $\{\sigma \mid \sigma\Phi = \Phi\} \subset \mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$.

In particular, in the case we are interested in, when $g = 2$, (K, Φ) is non-primitive if and only if K is biquadratic.

Definition 3.2.5 (CM abelian varieties)**.** Let A be an abelian variety of dimension g, defined over $\mathbb{C}$. we say that A is of CM type (K, Φ), if there is an ideal $\mathfrak{a}$ of K such that $A(\mathbb{C}) \cong \mathbb{C}^g/\Phi(\mathfrak{a})$.

In fact, such an A always have $\mathrm{End}(A) \cong \mathcal{O}_K$. In addition, the abelian variety given by the g-dimensional lattice $\mathbb{C}^g/\Phi(\mathfrak{a})$ is principally polarizable if and only if there exists a $\xi \in \mathcal{O}_K$, satisfying the following condition: (1) ξ is purely imaginary; (2) for each $\phi_i \in \Phi$, $\mathrm{Im}(\phi_i(\xi)) < 0$; (3) $\xi\mathfrak{D}_K = \mathfrak{a}\bar{\mathfrak{a}}$, where $\mathfrak{D}_K$ is the different of the CM field K.

Our goal is to characterize the principally polarized abelian varieties A with primitive CM type (K, Φ) via algebraic objects in K. In the elliptic curve case, we corresponded CM curves with the class group. In the general case, we need the Shimura class group of a CM field:

Definition 3.2.6 (The Shimura class group)**.** For a CM field K, the Shimura class group of a CM field, denoted as $\mathfrak{C}(K)$, is defined as

$$\mathfrak{C}(K) := \{(\mathfrak{a}, \alpha) \mid \mathfrak{a} \text{ fractional } \mathcal{O}_K\text{-ideal}, \mathfrak{a}\bar{\mathfrak{a}} = (\alpha), \alpha \in K \text{ totally positive}\}/\sim,$$

and we say $(\mathfrak{a}, \alpha) \sim (\mathfrak{b}, \beta)$ if there exists an $u \in K^\times$ satisfying $\mathfrak{b} = u\mathfrak{a}$ and $\beta = u\bar{u}\alpha$. The group law of $\mathfrak{C}(K)$ is given by component-wise multiplication.

For a CM type (K, Φ), denote by $S(K, \Phi)$ the set of isomorphism classes of principally polarizable abelian variety with CM by (K, Φ). Then, the best analog we can make from the main theorem of CM of elliptic curves is the following:

Theorem 3.2.7 (See also [Shi98, Koh08, BGL11])**.** There exists a transitive action of $\mathfrak{C}(K)$ on $S(K, \Phi)$. Let $H^\dagger$ be the Hilbert class field of $K^\dagger$, the reflex field of K. Then there exists a group homomorphism $\mathrm{Gal}(H^\dagger/K^\dagger) \to \mathfrak{C}(K)$.

As Kohel pointed out in [Koh08], the homomorphism $\mathrm{Gal}(H^\dagger/K^\dagger) \to \mathfrak{C}(K)$ need not be injective nor surjective. This leads to a weaker result compared to the elliptic curve case: If A is a principally polarized abelian variety with CM type (K, Φ), and suppose its Igusa invariants are (i_1, i_2, i_3), then the compositum $K^\dagger(i_1, i_2, i_3) \subseteq H^\dagger$ ([Spa94, Theorem 5.8]), but they do not necessarily needs to be equal.

Theorem 3.2.7 also suggests that $|\mathfrak{C}(K)| = |S(K, \Phi)|$, which could tell us the degree of the Igusa class polynomials if we can enumerate $|\mathfrak{C}(K)|$. This is given by the following theorem:

Theorem 3.2.8 ([BGL11, Theorem 3.1])**.** Let K be a primitive CM field with the totally real subfield K_0. Denote by $\mathrm{Cl}^+(K_0)$ the narrow class group of K_0, and $(\mathcal{O}_{K_0}^\times)^+$ the group of totally positive elements in $\mathcal{O}_{K_0}$.

Then the following sequence is exact:

$$1 \to (\mathcal{O}_{K_0}^\times)^+/N_{K/K_0}(\mathcal{O}_K^\times) \to \mathfrak{C}(K) \to \mathrm{Cl}(K) \to \mathrm{Cl}^+(K_0) \to 1,$$

where the three maps in the middle are defined by $u \mapsto (\mathcal{O}_K, u)$, $(\mathfrak{a}, \alpha) \mapsto \mathfrak{a}$, and the norm map, respectively.

In the case when $g = 2$, for a quartic CM field K, there is one CM type (up to conjugation) when K is cyclic and Galois, and two CM types when K is non-Galois. From Theorems 3.2.7 and 3.2.8, we have the following corollary which determines the degree of the Igusa class polynomials:

Corollary 3.2.9 ([Str14, Proposition 4.4], [BGL11, Corollary 3.3])**.** Let K be a primitive quartic CM field, and denote

$$\mathscr{N} = \left|(\mathcal{O}_{K_0}^{\times})^{+} / N_{K/K_0}(\mathcal{O}_K^{\times})\right| \cdot \frac{|\mathrm{Cl}(K)|}{|\mathrm{Cl}^{+}(K_0)|} = \left|\mathcal{O}_{K_0}^{\times} / N_{K/K_0}(\mathcal{O}_K^{\times})\right| \cdot \frac{|\mathrm{Cl}(K)|}{|\mathrm{Cl}(K_0)|}.$$

Then there are $\mathscr{N}$ and $2\mathscr{N}$ isomorphism classes of principally polarized abelian varieties with CM by K when K is Galois and non-Galois, respectively.

3.2.3 Canonical Lifting of Hyperelliptic Curves

Let k be a perfect field of characteristic $p > 0$, and let A_0 be an ordinary abelian variety over k.

Definition 3.2.10 (Canonical lifting)**.** Given A_0 as above, let $W(k)$ be the ring of infinite Witt vectors over k. We say that an abelian scheme A over $W(k)$ is a canonical lifting, if $A_k \cong A_0$ and the induced homomorphism $\mathrm{End}_{W(k)\text{-gr}}(A) \to \mathrm{End}_{k\text{-gr}}(A_0)$ is bijective. Here $A_k := A \times_{W(k)} k$ is the change of basis.

The first important fact is that canonical lifting exists:

Theorem 3.2.11 (Lubin, Serre and Tate [LST64]; see also Messing [Mes72], theorem V.3.3, p. 172)**.** For any ordinary abelian variety over k, there is a projective abelian $W(k)$-scheme A which is the canonical lifting of A_0.

Let $K = \mathrm{Frac}(W(K))$ be the fraction field. Then the geometric fiber A_K of the $W(K)$-scheme A is an abelian variety over K. We also call A_K the canonical lifting of A_0. A can be obtained by A_K via the Néron model.

The next issue is the Frobenius. The Frobenius in $W(k)$, denoted as $\sigma \in \mathrm{Aut}(W(k))$, is defined by sending each component of the Witt vector to its p-th power. We thus have the relative Frobenius for abelian varieties on $W(k)$ and on k.

Proposition 3.2.12 ([Mes72, Corollary A.1.2, p. 177]). Let A be a $W(k)$-scheme and A_0 be an ordinary abelian variety over k such that $A_k \cong A_0$. A is a canonical lifting of A_0 if and only if the relative Frobenius $F_0 : A_0 \to A_0^{(p)}$ lifts to a separable isogeny $F : A \to A^\sigma$.

3.2.4 Theta Functions

We shall look at two sides of theta functions: the analytic theta functions, which is defined for abelian varieties defined over $\mathbb{C}$; and the algebraic theta functions, which can be defined for any abelian variety. We will also look at how they are connected.

To define the analytic theta function, let A be an abelian variety of dimension g over $\mathbb{C}$. Then A is associated to a period matrix $\Omega \in \mathfrak{H}^g$, that is, $A \cong \mathbb{C}^g/(\mathbb{Z}^g + \Omega\mathbb{Z}^g)$. Here $\mathfrak{H}^g$ is the "g-dimensional upper half-plane", that is,

$$\mathfrak{H}^g := \{\Omega \in \mathrm{Mat}(g \times g, \mathbb{C}) \mid \text{The imaginary part of } \Omega \text{ is positive definite}\}.$$

Definition 3.2.13 (Analytic theta functions). Let $\epsilon_1, \epsilon_2 \in \mathbb{Z}^g$. And let $l \in \mathbb{Z}$. Then the analytic theta function is defined as

$$\theta_l \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \end{bmatrix} (z, \Omega) := \sum_{n \in \mathbb{Z}^g} \exp\left[\pi i\Big(n + \frac{\epsilon_1}{l}\Big)^t \Omega \left(n + \frac{\epsilon_1}{l}\right) + 2\pi i\Big(n + \frac{\epsilon_1}{l}\Big)^t \left(z + \frac{\epsilon_2}{l}\right)\right].$$

We will need a similar construction which works on abelian varieties over any field. We will do so by introducing algebraic theta structures. A complete treatise of this topic can be found in the work of Mumford [Mum66].

As usual, suppose A is an abelian variety of dimension 2 over a field k, and $\mathscr{L}$ is an ample line bundle on A of degree d. Let $A^\vee := \mathrm{Pic}^0(A)$ be the dual abelian variety of A. Then there is a homomorphism associated to $\mathscr{L}$, given by $\phi_{\mathscr{L}} : A \to A^\vee$, $x \mapsto \langle \tau_x^* \mathscr{L} \otimes \mathscr{L}^{-1} \rangle$, where $\tau_x : A \to A$ is the "shift by x homomorphism". Now we denote $K(\mathscr{L}) = \ker(\phi_{\mathscr{L}})$ (so that if $\mathrm{char}(k) \not| \, d$, $K(\mathscr{L})$ has cardinality d^2), and define $G(\mathscr{L}) := \{(x, \varphi) \mid x \in K(\mathscr{L}), \varphi : \mathscr{L} \xrightarrow{\sim} \tau_x^* \mathscr{L}\}$. We can define a group structure on $G(\mathscr{L})$ by $(x, \varphi) \cdot (y, \psi) := (x + y, \tau_y \varphi \circ \psi)$.

On the other hand, define $\delta = (\delta_1, \delta_2) \in \mathbb{Z}^2_{>0}$ where $\delta_1 \mid \delta_2$. Let $Z(\delta) := \mathbb{Z}/\delta_1\mathbb{Z} \times \mathbb{Z}/\delta_2\mathbb{Z}$ be a finite group, and $Z(\delta)^D$ be the Cartier dual of $Z(\delta)$. We then define $K(\delta) := Z(\delta) \times Z(\delta)^D$. Note that for each $\mathscr{L}$ there is a unique δ such that $K(\delta)$ is isomorphic to $K(\mathscr{L})$. We also define $H(\delta) := \mathbb{G}_{m,k} \times Z(\delta) \times Z(\delta)^D$ equipped with the group law $(\alpha_1, x_1, l_1) \cdot (\alpha_2, x_2, l_2) := (\alpha_1\alpha_2 l_2(x_1), x_1 + x_2, l_1 l_2)$. Under all these settings we can define theta structures as follows:

Definition 3.2.14. With notation as above, we define the theta structure as a homomorphism $\Theta_\delta : H(\delta) \to G(\mathscr{L})$ which makes the diagram below commutative.

$$\begin{tikzcd}
0 \arrow[r] & \mathbb{G}_{m,k} \arrow[r] \arrow[d, equal] & H(\delta) \arrow[r] \arrow[d, "\Theta_\delta"] & K(\delta) \arrow[r] \arrow[d, "\bar{\Theta}_\delta"] & 0 \\
0 \arrow[r] & \mathbb{G}_{m,k} \arrow[r] & G(\mathscr{L}) \arrow[r, "\kappa"] & K(\mathscr{L}) \arrow[r] & 0
\end{tikzcd}$$

Furthermore, suppose $\mathscr{L}$ is symmetric (which means there exists $\psi : [-1]^*\mathscr{L} \xrightarrow{\sim} \mathscr{L}$). We say that Θ_δ is symmetric if $\delta_{-1} \circ \Theta_\delta = \Theta_\delta \circ D_{-1}$, where $D_{-1} : H(\delta) \to H(\delta)$, $(\alpha, x, l) \mapsto (\alpha, -x, l^{-1})$ and $\delta_{-1} : G(\mathscr{L}) \to G(\mathscr{L})$, $(x, \varphi) \mapsto (-x, \tau_\varphi)$, with $\tau_\varphi := \tau_x^* \psi^{-1} \circ [-1]^* \varphi \circ \psi$. If both $\mathscr{L}$ and Θ_δ are symmetric, we call the data $(A, \Theta_\delta, \mathscr{L})$ an abelian variety with a δ-marking.

Under this setting, Mumford [Mum66] proved that the global section $\Gamma(A, \mathscr{L})$ is an irreducible $G(\mathscr{L})$-module. Together with the theta structure Θ_δ, we will get a unique embedding $A \hookrightarrow \mathbb{P}(\Gamma(A, \mathscr{L}))$, and the theta structure canonically defined $\mathbb{P}(\Gamma(A, \mathscr{L})) \xrightarrow{\sim} \mathbb{P}(V(\delta))) \xrightarrow{\sim} \mathbb{P}_k^{d-1}$, where $V(\delta)$ is the vector space with basis $Z(\delta)$.

Definition 3.2.15 (Theta null point)**.** Let $(A, \Theta_\delta, \mathscr{L})$ be a marked abelian variety. We call the theta null point the image of e, the identity of A, in the projective space $\mathbb{P}_k^{d-1}$ via the canonical embedding described above.

Now we move to the situation about which we actually care: Let $\delta = \overline{nl} = (nl, nl)$, where $n = 2^v$ and l is a prime. For an $\overline{nl}$-marking $(A, \Theta_{\overline{nl}}, \mathscr{L})$, we associate it with its canonical theta null point $(a_i)_{i \in Z(\overline{nl})}$. It is well known (see for example, [Mum66]), that the coefficients $(a_i)_{i \in Z(\overline{nl})}$ of a theta null point have the following restrictions.

Theorem 3.2.16 (Mumford, [Mum66])**.** Let $(a_i)_{i \in Z(\overline{nl})}$ be a theta null point, as defined above. Then:

1. (Symmetry inherited from $\Theta_{\overline{nl}}$) For all $i \in Z(\overline{nl})$, $a_i = a_{-i}$.
2. (Riemann's relations) Let (v_j, w_j, x_j, y_y), $j = 1, 2$ be in $Z(\overline{nl})^4$ such that the two quadruples $(v_1 + w_1, v_1 - w_1, x_1 + y_1, x_1 - y_1)$ and $(v_2 + w_2, v_2 - w_2, x_2 + y_2, x_2 - y_2)$ only differ by permutation. Let $\chi \in Z(\bar{2})^D$. Then we have

$$\sum_{t \in Z(\bar{2})} \chi(t) a_{v_1+t} a_{w_1+t} \sum_{s \in Z(\bar{2})} \chi(s) a_{x_1+s} a_{y_1+s} = \sum_{t \in Z(\bar{2})} \chi(t) a_{v_2+t} a_{w_2+t} \sum_{s \in Z(\bar{2})} \chi(s) a_{x_2+s} a_{y_2+s}.$$

Another desirable property is that the equations above actually characterize the moduli space of abelian varieties with $\overline{nl}$-covering:

Theorem 3.2.17 ([Mum67, p. 87])**.** Suppose $n = 2^v \geq 8$. Let $\mathscr{M}_{\overline{nl}}$ be the locus of theta null points of abelian varieties with $\overline{nl}$-marking, and let $\bar{\mathscr{M}}_{\overline{nl}}$ be the closed projective subvariety of $\mathbb{P}_k^{d-1}$ defined by all the equations in Theorem 3.2.16. Then $\mathscr{M}_{\overline{nl}}$ maps non-isomorphic $\overline{nl}$-markings to different points on $\mathscr{M}_{\overline{nl}}$, and $\mathscr{M}_{\overline{nl}}$ is an open subvariety of $\bar{\mathscr{M}}_{\overline{nl}}$.

3.3 The Main Algorithm

In this section, we describe the major steps of the principal ideal generator algorithm. After the algorithm, we give comments on each step, together with a pointer on where the steps will be explained. The validity of the steps of the algorithm will be discussing following the subsections describing each step. The complexity of the algorithm will be focused on Section

Algorithm 3.3.1 (The main algorithm)**.**
Input: A quadratic CM field K, with the real quadratic field K_0.
Output: The Igusa polynomials $h_{K,n}(x)$, where $n = 1, 2, 3$ of K.

(1) Search for the smallest prime power p^r such that there exists a hyperelliptic curve C defined over $\mathbb{F}_{p^r}$ with $\mathrm{End}(\mathrm{Jac}(C)) = \mathcal{O}_K$.

(2) Find a $(i_1, i_2, i_3) \in \mathbb{F}_{p^r}^3$ such that the curve $C/\mathbb{F}_{p^r}$ with the Igusa invariant (i_1, i_2, i_3) has endomorphism ring $\mathcal{O}_K$.

(3) Compute the canonical lift $\tilde{C}/\mathbb{Q}_{p^r}$ of C.

(4) Compute the Igusa invariants $(\tilde{i_1}, \tilde{i_2}, \tilde{i_3}) \in \mathbb{Q}_{p^r}$ of $\tilde{C}$.

    Compute the minimal polynomials $\{\widetilde{h_{K,1}}, \widetilde{h_{K,2}}, \widetilde{h_{K,3}}\} \subset \mathbb{Q}[x]$ of $\{\tilde{i_1}, \tilde{i_2}, \tilde{i_3}\}$.

(5) If all the degrees of $\widetilde{h_{K,i}}$ equal to the expected degree, which can be computed from K, return $\{\widetilde{h_{K,1}}, \widetilde{h_{K,2}}, \widetilde{h_{K,3}}\}$ and terminate. Otherwise, $\{\widetilde{h_{K,1}}, \widetilde{h_{K,2}}, \widetilde{h_{K,3}}\}$ are factors of h_1, h_2, h_3. Go back to step (2) and find another set of (i_1, i_2, i_3) to find other factors.

Below are some comments and rationales for the steps in Algorithm 3.3.1, together with pointers to the relevant sections.

(1) As we will explain in Section 3.4.1, the characteristic p depends on the splitting condition in the number field K, and the extension degree r depends on both the class group of K and $H^\dagger$, the Hilbert class field of the reflex field of K.

(2) Mestre gave an algorithm in [Mes91] to obtain a curve C from the Igusa invariants. The remaining is a two-step check: To test if $\mathrm{Jac}(C)$ has CM by an order in K, one can verify by point counting; to test if $\mathrm{Jac}(C)$ has CM by K, one can use the ideas of [EL10, FL08, Spr19]. More will be discussed in 3.4.

(3) This is the core of the main algorithm. As will be explained in Section 3.5, we will first extract the 2-theta null point of C from its hyperelliptic equation, then the $2p$-theta null point of C by solving the Riemann equations as in Theorem 3.2.17, Finally, to compute the $2p$-theta null point of $\tilde{C}$, one solves the Artin-Schreier equation induced from Theorem 3.5.9.

(4) This is in Section 3.6. The Igusa invariants of $\tilde{C}$ can be derived from the $2p$-theta null points by reverting step (3) above. Then the minimal polynomial can be constructed via the LLL-algorithm, as described in [GHK$^+$06], if the degree and an upper bound of the coefficients of the Igusa class polynomial are known.

(5) The expected degree of the Igusa class polynomials are given by Corollary 3.2.9.

3.4 Finding a Hyperelliptic Curve over a Finite Field with CM by a Maximal Order

This section discusses steps 1 and 2 of the main algorithm (Algorithm 3.3.1 in greater detail. The goal is to find a hyperelliptic curve C over a finite field whose Jacobian has the correct CM. We describe the algorithm which we implemented in Algorithm 3.4.1. Then we will discuss the validity and give some possible improvements.

Algorithm 3.4.1 (Finding a genus 2 curve whose Jacobian is the prescribed maximal order)**.**

Input: A quartic CM field K
Output: A finite field $\mathbb{F}_q = \mathbb{F}_{p^r}$; a hyperelliptic curve $\mathcal{C}$ defined over $\mathbb{F}_q$, satisfying $\mathrm{End}(\mathrm{Jac}(\mathcal{C})) \cong \mathcal{O}_K$.

```
1: Find a prime $p$ such that there exists abelian surfaces $A$ defined over $\overline{\mathbb{F}_p}$ with $\mathrm{End}(A) \cong \mathcal{O}_K$.
2: For the $p$ from step 1, find the smallest $q = p^r$ such that there exists abelian surfaces $A$ defined over $\mathbb{F}_q$ with $\mathrm{End}(A) \cong \mathcal{O}_K$.
3: while $\mathcal{C}$ is not found do
4:     for $(i_1, i_2, i_3)$ in $\mathbb{F}_q^3$ do
5:         Set $\mathcal{C}$ as the hyperelliptic curve over $\mathbb{F}_q$ with Igusa invariants $(i_1, i_2, i_3)$.
```

```
6:         Discard $\mathcal{C}$ if $\mathrm{End}(\mathrm{Jac}(\mathcal{C})) \otimes_{\mathbb{Z}} \mathbb{Q} \neq K$.
7:         Discard $\mathcal{C}$ if if $\mathrm{End}(\mathrm{Jac}(\mathcal{C})) \subsetneq \mathcal{O}_K$.
8:         if $\mathcal{C}$ not discarded then
9:             Return $(p^r, \mathcal{C})$.
10:        end if
11:    end for
12: end while
```

For the rest of the section, we will walk through implementation issues on the key steps of the algorithm and demonstrate the validity of Algorithm 3.4.1. The three key steps will be: (1) Finding a finite field where hyperelliptic Jacobian with CM over K could possibly be found (Section 4.4.1); (2) Over the finite field, finding hyperelliptic Jacobians with the right endomorphism *algebra* (Section 4.4.2); and (3) From the hyperelliptic Jacobians in the last steps, pick the right one with the right endomorphism *ring* (Section 4.4.3).

3.4.1 Finding suitable finite field

Step 2 involves finding the smallest possible finite field $\mathbb{F}_q$ such that there are hyperelliptic curves $\mathcal{C}$ defined over $\mathbb{F}_q$ which has CM by the maximal order $\mathcal{O}_K$. The characteristic of the field $\mathbb{F}_q$ will be determined by Theorem 3.4.2; and the extension degree of the finite field will be determined by Lemma 3.4.3.

We start from the opposite side of the construction: Suppose there is a hyperelliptic curve $\widetilde{\mathcal{C}}$ of genus 2 defined over a number field L such that $\mathrm{End}(\widetilde{\mathcal{C}}) \cong \mathcal{O}_K$, and $\mathfrak{p}$ is a prime ideal in L. We are interested in the reduction of $\tilde{\mathrm{C}}$ modulo $\mathfrak{p}$, and in our application, we need $\widetilde{\mathcal{C}}$ reduce to a curve $\mathcal{C}$ whose Jacobian is ordinary. Goren gave a complete characterization in [Gor97] on the reduction of an abelian surface over a number field, which we summarize the relevant part as the following theorem:

Theorem 3.4.2 (Ordinary reduction of CM abelian surfaces, [Gor97, Theorems 1 and 2])**.** Suppose that $\widetilde{A}$ is an abelian surface defined over a number field L, with CM by $\mathcal{O}_K$ for a primitive quartic CM field K. Let M be the compositum of K and L. Let $\mathfrak{P}$ be a prime in M, $\mathfrak{p} = \mathfrak{P} \cap \mathcal{O}_K$, and $(p) = \mathfrak{p} \cap \mathbb{Z}$. Suppose p is unramified in K, and let A be the reduction of $\widetilde{A}$ modulo $\mathfrak{P}$.

(a) If K is a cyclic extension over $\mathbb{Q}$, then A is ordinary and simple if and only if $p\mathcal{O}_K = \mathfrak{p}_1\mathfrak{p}_2\overline{\mathfrak{p}_1\mathfrak{p}_2}$, where $\overline{\mathfrak{p}}$ means the element-wise complex conjugation on $\mathfrak{p}$.

(b) If K is a non-Galois extension over $\mathbb{Q}$, take K' to be the Galois closure of K, and fix a embedding $K' \hookrightarrow \overline{\mathbb{Q}}$. We have $\mathrm{Gal}(K'/\mathbb{Q}) \cong D_4$. Choose suitable $x, y \in D_4$ such that

$D_4 \cong \langle x, y \ : \ x^2, y^4, xyxy \rangle$, K is the fixed field of x, and suppose the CM type of $\widetilde{A}$ is $\Phi = \{1, y\}$.

Under the setting above, for the reduction of $\widetilde{A}$ modulo $\mathfrak{P}$ to be ordinary, one of the following two cases have to be true:

(i) $p\mathcal{O}_{K'} = \mathfrak{p}_1\mathfrak{p}_y\mathfrak{p}_{y^2}\mathfrak{p}_{y^3}\mathfrak{p}_x\mathfrak{p}_{xy}\mathfrak{p}_{xy^2}\mathfrak{p}_{xy^3}$ splits completely (here, for $g \in D_4$, we denote $\mathfrak{p}_g := g\mathfrak{p}$ to be the action by g), the decomposition group of $\mathfrak{p}_1 = \{1\}$, and

$$p\mathcal{O}_K = (\mathfrak{p}_1\mathfrak{p}_x)(\mathfrak{p}_y\mathfrak{p}_{xy})(\mathfrak{p}_{y^2}\mathfrak{p}_{xy^2})(\mathfrak{p}_{y^3}\mathfrak{p}_{xy^3}).$$

(ii) $p\mathcal{O}_{K'} = \mathfrak{p}_1\mathfrak{p}_y\mathfrak{p}_{y^2}\mathfrak{p}_{y^3}$, the decomposition group of $\mathfrak{p}_1 = \{1, xy^3\}$, and $p\mathcal{O}_K = (\mathfrak{p}_1\mathfrak{p}_{y^3})(\mathfrak{p}_y\mathfrak{p}_{y^2})$.

From Theorem 3.4.2, we can see that for any quartic CM field K, a necessary condition to have a curve defined over $\mathbb{F}_{p^r}$ whose Jacobian has CM by $\mathcal{O}_K$ is that p splits as either $\mathfrak{p}\overline{\mathfrak{p}}$ or $\mathfrak{p}_1\mathfrak{p}_2\overline{\mathfrak{p}_1\mathfrak{p}_2}$.

Now we will need to answer the opposite side of the question: Suppose p is a prime that splits completely in a CM field K. We need to find the smallest finite field extension $\mathbb{F}_{p^r}$ such that there exists an abelian variety A over $\mathbb{F}_{p^r}$ with CM by $\mathcal{O}_K$. We have the following lemma which gives characterizes how to find such a finite field.

Lemma 3.4.3. Suppose K is a primitive quartic CM field, $K \neq \mathbb{Q}(e^{2\pi i/5})$, and p is a prime which splits as in Theorem 3.4.2, in the sense that an abelian variety with CM by $\mathcal{O}_K$ has ordinary reduction. Suppose $\mathbb{F}$ is a field of characteristic p such that there exists a hyperelliptic curve over $\mathbb{F}$ whose Jacobian has CM by $\mathcal{O}_K$.

(a) The lower bound of the size of $\mathbb{F}$ is as follows:

(i) If p splits into two conjugate primes $\mathfrak{p}, \overline{\mathfrak{p}}$ in K, and suppose $\mathfrak{p}$ has order r in the class group $\mathrm{Cl}(K)$ of K, then $\mathbb{F}$ contains $\mathbb{F}_{p^r}$.

(ii) If p splits completely into $\mathfrak{p}_1, \overline{\mathfrak{p}_1}, \mathfrak{p}_2, \overline{\mathfrak{p}_2}$ in K, and suppose $\mathfrak{p}_1\mathfrak{p}_2$ and $\mathfrak{p}_1, \overline{\mathfrak{p}_2}$ have orders r_1 and r_2 in $\mathrm{Cl}(K)$, respectively, then $\mathbb{F}$ contains either $\mathbb{F}_{p^{r_1}}$ or $\mathbb{F}_{p^{r_2}}$.

(b) The upper bound of the size of $\mathbb{F}$ is as follows. Suppose $K^\dagger$ is the reflex field of K, and $H^\dagger$ is the Hilbert class field of $K^\dagger$. Suppose that $f_1, \cdots, f_l$ are the inertia degrees of prime ideals over p in $H^\dagger$, then $\mathbb{F}$ is contained in $\mathbb{F}_{p^{f_i}}$ for some i.

Proof. (a) It turned out that this part of the lemma is at least implemented without proof in Kohel's Echidna library ([Koh]). We will prove this part for the sake of completeness.

Suppose A is an abelian variety over $\mathbb{F}$ with CM by $\mathcal{O}_K$ Fix an embedding of $\mathrm{End}_{\overline{\mathbb{F}}}(A)$ in K, we will have the $\mathbb{F}$-Frobenius endomorphism $\pi \in \mathcal{O}_K$, and from the Honda-Tate theory, $\pi\overline{\pi} = |\mathbb{F}|$. In case (i), if $p\mathcal{O}_K = \mathfrak{p}\overline{\mathfrak{p}}$, then π is either in $\mathfrak{p}$ or $\overline{\mathfrak{p}}$. Furthermore, if $|\mathbb{F}| = p^r$, then since p does not divide π, π is either in $\mathfrak{p}^r$ or $\overline{\mathfrak{p}}^r$, and by comparing the norm, π is the generator of either $\mathfrak{p}^r$ or $\overline{\mathfrak{p}}^r$, which enforces both of them to be principal ideals, and r is a multiple of the order of $\mathfrak{p}$ in $\mathrm{Cl}(K)$. Similarly, in case (ii), $p\mathcal{O}_K = \mathfrak{p}_1\overline{\mathfrak{p}_1}\mathfrak{p}_2\overline{\mathfrak{p}_2}$, π must be contained in two non-conjugate prime ideals over p, which implies π is one of $(\mathfrak{p}_1\mathfrak{p}_2)^r$, $(\mathfrak{p}_1\overline{\mathfrak{p}_2})^r$, $(\overline{\mathfrak{p}_1}\mathfrak{p}_2)^r$, or $(\overline{\mathfrak{p}_1\mathfrak{p}_2})^r$. And this enforces the extension degree r of $\mathbb{F}$ must divide the order of $\mathfrak{p}_1\mathfrak{p}_2$ or $\mathfrak{p}_1\overline{\mathfrak{p}_2}$ in $\mathrm{Cl}(K)$.

(b) From [Spa94, Theorem 5.8], it is known that if $\widetilde{A}$ is a principally polarized abelian variety defined over $\overline{\mathbb{Q}}$ which has CM by K, and let $(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$ be the Igusa invariants of $\widetilde{A}$, then the field $K^\dagger(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$ is contained in the Hilbert class field $H^\dagger$. Suppose $\mathfrak{P}$ is a prime ideal over p in $K^\dagger(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$ of inertia degree f, then $\widetilde{A}$ has a ordinary reduction over the residue field $\mathbb{F}_{p^f}$, with Igusa invariants (i_1, i_2, i_3) in the residue field. We claim that (i_1, i_2, i_3) corresponds to a hyperelliptic curve C with $\mathrm{Aut}(C) \cong \mathbb{Z}/2\mathbb{Z}$. In [Igu60, section 8], all the possible $\mathrm{Aut}(C)$ are listed, and from [Igu60, Lemma 9], among those possibilities, if $\mathrm{Aut}(C)$ contains $(\mathbb{Z}/2\mathbb{Z})^2$ as a subgroup, then C has split Jacobian (i.e. $\mathrm{Jac}(C)$ is isogenous to a product of elliptic curves). This cannot happen since $\mathrm{Jac}(C)$, as a reduction, has CM by $\mathcal{O}_K$. For the cases where $(\mathbb{Z}/2\mathbb{Z})^2$ is not a subgroup of $\mathrm{Aut}(C)$, $\mathrm{Aut}(C)$ contains an element of order 3 or 5. This implies that K is either a biquadratic field of $\mathbb{Q}(e^{2\pi i/5})$, which we have already excluded.

When $(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3) \in \mathbb{F}_{p^f}$ corresponds to a curve satisfying $\mathrm{Aut}(C) \cong \mathbb{Z}/2\mathbb{Z}$, Mestre gave an algorithm in [Mes91] which recovers the hyperelliptic curve C from the Igusa invariants, defined over the same field $\mathbb{F}_{p^f}$. And we know that $\mathrm{End}(\mathrm{Jac}(C)) \supset \mathrm{End}(\widetilde{A}) \cong \mathcal{O}_K$ as desired.

□

3.4.2 Finding a hyperelliptic Jacobian with the correct endomorphism algebra

Now, suppose that given a primitive CM field, the finite field $\mathbb{F}_q = \mathbb{F}_{p^r}$ is chosen according to the conditions in Lemma 3.4.3. In this subsection, we will explain step 5 in Algorithm 3.4.1, which finds a curve C such that $\mathrm{End}(\mathrm{Jac}(C)) \otimes_{\mathbb{Z}} \mathbb{Q} \cong K$. The criteria for such C are defined in Lemma 3.4.4, for which the case wheen $q = p^1$ is described in [EL10].

Lemma 3.4.4. Suppose K is a primitive quartic CM field and $\mathbb{F}_q = \mathbb{F}_{p^r}$ is a finite field satisfying the conditions in Lemma 3.4.3. For a curve C defined over the finite field $\mathbb{F}_q$, denote $N_i = |C(\mathbb{F}_{q^i})|$, and $N_J = |\mathrm{Jac}(C)(\mathbb{F}_q)|$. If $\mathrm{Jac}(C)$ has CM by $\mathcal{O}_K$, there are at most 2 of 4 possible values for the pairs (N_1, N_2) and (N_1, N_J).

Proof. According to the proof of Lemma 3.4.3, for a fixed primitive CM field K, there are either 2 or 4 possible $\pi \in \mathcal{O}_K$ elements satisfying $\pi\overline{\pi} = q$, which are generators of the prime ideals $(\mathfrak{p})^r$ (or either $(\mathfrak{p}_1\mathfrak{p}_2)^r$ or $(\mathfrak{p}_1\overline{\mathfrak{p}_2})^r$), depending on whether p splits into 2 or 4 primes in K. And we can compute from each possible π its minimal polynomial, which will be of the form $f(t) = t^4 + a_1t^3 + a_2t^2 + qa_1t + q^2$. Therefore, there are only 2 or 4 possible minimal polynomials for the q-Frobenius endomorphism a $\mathrm{Jac}(C)$ with CM by $\mathcal{O}_K$.

If $\mathrm{Jac}(C)$ has CM by $\mathcal{O}_K$, the zeta function associated with C is

$$Z(C,t) = \exp\left(\sum_{m=1}^{\infty} N_m \frac{t^m}{m}\right) = \frac{t^4 f(1/t)}{(1-t)(1-qt)} = \frac{1 + a_1t + a_2t^2 + qa_1t^3 + q^2t^4}{(1-t)(1-qt)}.$$

Matching the t and t^2-coeffiecients of the zeta functions, we get

$$N_1 = 1 + q + a_1;$$
$$\frac{1}{2}N_1^2 + \frac{1}{2}N_2^2 = (1 + q + q^2) + (1+q)a_1 + a_2.$$

Rearranging the terms, we get $N_2 = 1 + q^2 - a_1^2 + 2a_2$. It is clear that (N_1, N_2) uniquely determines (a_1, a_2). This means to find C such that $\mathrm{Jac}(C)$ has CM by $\mathcal{O}_K$, it is a necessary condition that

$$(|C(\mathbb{F}_q)|, |C(\mathbb{F}_{q^2})|) = (1 + q + a_1, 1 + q^2 - a_1^2 + 2a_2).$$

Equivalently, as we have $|\mathrm{Jac}(C)(\mathbb{F}_q)| = N(\pi - 1)$. Since this is the constant term of $f(t-1)$, we have $|\mathrm{Jac}(C)(\mathbb{F}_q)| = 1 + q^2 - (1+q)a_1 + a_2$. Again one readily checks that $(N_1, |\mathrm{Jac}(C)(\mathbb{F}_q)|)$ uniquely determines (a_1, a_2) as well, so it is also a necessary condition that

$$(|C(\mathbb{F}_q)|, |\mathrm{Jac}(C)(\mathbb{F}_{q^2})|) = (1 + q + a_1, 1 + q^2 - (1+q)a_1 + a_2).$$

In summary, given K and $\mathbb{F}_q$, we will obtain 2 or 4 possible Weil-q numbers π and values of (N_1, N_2) (or equivalently, $(N_1, |\mathrm{Jac}(C)(\mathbb{F}_q)|)$), which are the necessary conditions for $|\mathrm{Jac}(C)|$ to have K as the endomorphism algebra. □

To summarize what we have done so far, Algorithm 3.4.1 loops over all possible Igusa invariants $(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$, and uses Mestre's algorithm in [Mes91] to recover a curve C over $\mathbb{F}_q$. We have a finite set of necessary condition on the number of points (N_1, N_2) or (N_1, N_J) for C. If the conditions for C is satisfied, then $\mathrm{End}(\mathrm{Jac}(C)) \otimes_{\mathbb{Z}} \mathbb{Q} \cong K$, and we will proceed to the next section to see if $\mathrm{End}(\mathrm{Jac}(C)) \cong \mathcal{O}_K$.

3.4.3 Finding a hyperelliptic Jacobian with the correct endomorphism ring

Suppose C is a hyperelliptic curve such that $\mathrm{Jac}(C)$ corresponds to the Weil-q number π. Then, after identifying $\mathrm{End}(\mathrm{Jac}(C)) \hookrightarrow \mathcal{O}_K$, π and $\overline{\pi}$ corresponds to the q-Frobenius and Verschiebung endomorphism, respectively, and this implies $\mathbb{Z}[\pi, \overline{\pi}] \subseteq \mathrm{End}(\mathrm{Jac}(C))$. Therefore, we need to determine whether $[\mathcal{O}_K : \mathrm{End}(\mathrm{Jac}(C))] = 1$, knowing a priori that this index divides $[\mathcal{O}_K : \mathbb{Z}[\pi, \overline{\pi}]]$.

Eisenträger and Lauter discussed possible approaches to determine $[\mathcal{O}_K : \mathrm{End}(\mathrm{Jac}(C))] = 1$ in [EL10, Section 6], with some additional assumptions such as K_0, the real subfield of K, has class number 1. A closer discussion, with the class number requirements removed, is in [FL08].

The idea in [FL08] is to find a set of generator $\{\alpha_i\}$, so that $\mathbb{Z}[\pi, \overline{\pi}\{\alpha_i\}] = \mathcal{O}_K$ and each α_i is of the form $(\pi^k - 1)/\ell$ or $(s_0 + s_1\pi + s_2\pi^2 + s_3\pi^3)/\ell^d$. In the former case, $(\pi^k - 1)/\ell \in \mathrm{End}(\mathrm{Jac}(C))$ is equivalent to $\mathrm{Jac}(C)[\ell]$ being contained in $\mathbb{F}_{p^k}$; and in the latter case, $(s_0 + s_1\pi + s_2\pi^2 + s_3\pi^3)/\ell^d \in \mathrm{End}(\mathrm{Jac}(C))$ is equivalent to $s_0 + s_1\pi + s_2\pi^2 + s_3\pi^3$ acts as a zero map on $\mathrm{Jac}(C)[\ell^d]$. Both can be tested via probabilistic methods.

Although we will be using the implementation of endomorphism ring computation in Echidna [Koh], which is based on the idea of [EL10] and [FL08], these algorithms could be inefficient and even infeasible, since for each prime divisor ℓ of $[\mathcal{O}_K : \mathbb{Z}[\pi, \overline{\pi}]]$, the complexity of the algorithm depends on the underlying field of the ℓ^d-torsion of $\mathrm{Jac}(C)$ since we need to work on the arithmetic of this field. In the worst case, the complexity can reach $\tilde{O}(q^{18})$ ([BGL11]).

Finally, in the special case when $\mathrm{Jac}(C)$ has ream multiplication (RM), i.e. $\mathrm{End}(\mathrm{Jac}(C)) \cap K_0 = \mathcal{O}_{K_0}$, Springer proposed in [Spr19] a subexponential algorithm in computing $\mathrm{Jac}(C)$, which gives substantial improvements especially when there are large prime factors in the index $[\mathcal{O}_K : \mathrm{End}(\mathrm{Jac}(C))]$.

3.4.4 Discussing some potential improvements

Indeed, the triple loop in step 4 is quite hopeless when q gets moderately large (probably around 1000). For more hope, it has been suggested that one starts from a curve such that $\mathrm{Jac}(C)$ has the right endomorphism algebra. Then, for each ℓ dividing $[\mathcal{O}_K : \mathrm{Jac}(C)]$, one can attempt to take an (ℓ, ℓ)-isogeny path until arriving at an isogenous Jacobian $\mathrm{Jac}(C')$ whose endomorphism ring is maximal at ℓ (so $\ell \nmid [\mathcal{O}_K : \mathrm{Jac}(C)]$). In the elliptic case, this is possible, since the ℓ-isogeny graph has a "volcano" structure, as stated in Kohel's thesis [Koh96]. However, the isogeny volcano structure no longer exists in dimension 2, since the (ℓ, ℓ)-isogeny graph might not even be connected. This approach is tried and discussed in

[BGL11], and then in [RL13]. Later, in the special case when $\mathrm{Jac}(C)$ has maximal RM, the (ℓ, ℓ)-graph structure is better understood by Brooks, Jetchev, Wesolowski [BJW17].

3.5 Computing the Canonical Lift of a Hyperelliptic Jacobian over a Finite Field

This is the core of the algorithm. We will split the entire procedure into three subsections. The first part deals with computing the 2-theta null points over the finite field from the hyperelliptic equation. This step is standard and implemented in various packages, such as Echidna by Kohel [Koh]. The second part deals with the $2p$-theta null points over $\mathbb{F}_{p^r}$, and it involves using Gröbner systems to solve systems of Riemann equation. Finally, the third part involves computing the $2p$-theta null points over $\mathbb{Q}_{p^r}$. This involves solving a system of Artin-Schreier equations, and is where the canonical lift essentially happens.

3.5.1 Computing the 2-theta Null Points over $\mathbb{F}_{p^r}$

Suppose that $C : y^2 = f(x)$ is a hyperelliptic curve of genus 2 defined over $\mathbb{F}_q$, and pass to a finite field extension, suppose that C can be written in the Rosenhein form $C : y^2 = \prod_{i=1}^{5}(x - \alpha_i)$. Then the 2-theta null points of $A = \mathrm{Jac}(C)$ can be computed via the following proposition:

Proposition 3.5.1 (The implementation of Echidna [Koh]). Suppose $C : y^2 = \prod_{i=1}^{5}(x - \alpha_i)$ is a hyperelliptic curve over a field of odd characteristic. Then $(a_{00}, a_{01}, a_{10}, a_{11})$, computed via the steps below, gives a 2-theta null point of $\mathrm{Jac}(C)$.

Step 1: First, calculate an intermediate vector $(u_{00}, u_{01}, u_{10}, u_{11})$, where

$$u_{00} = 1; \quad u_{01} = \sqrt{\frac{(\alpha_1 - \alpha_2)(\alpha_1 - \alpha_4)}{(\alpha_1 - \alpha_3)(\alpha_1 - \alpha_5)}}; \qquad u_{11} = \sqrt{\frac{(\alpha_1 - \alpha_2)(\alpha_2 - \alpha_5)(\alpha_3 - \alpha_4)}{(\alpha_1 - \alpha_3)(\alpha_2 - \alpha_4)(\alpha_3 - \alpha_5)}};$$

$$u_{10} = \sqrt{\frac{(\alpha_1 - \alpha_4)(\alpha_2 - \alpha_5)(\alpha_3 - \alpha_4)}{(\alpha_1 - \alpha_5)(\alpha_2 - \alpha_4)(\alpha_3 - \alpha_5)}}.$$

Step 2: Using the intermediate vector $(u_{00}, u_{01}, u_{10}, u_{11})$, we can solve the following system of equations to get the 2-theta null points $(a_{00}, a_{01}, a_{10}, a_{11})$ (here t is a constant):

$$\begin{aligned}
(a_{00} + a_{01} + a_{10} + a_{11})^2 &= t(u_{00} + u_{01} + u_{10} + u_{11});\\
(a_{00} - a_{01} + a_{10} - a_{11})^2 &= t(u_{00} - u_{01} + u_{10} - u_{11});\\
(a_{00} + a_{01} - a_{10} - a_{11})^2 &= t(u_{00} + u_{01} - u_{10} - u_{11});\\
(a_{00} - a_{01} - a_{10} + a_{11})^2 &= t(u_{00} - u_{01} - u_{10} + u_{11}).
\end{aligned}$$

Proof of the validity of Proposition 3.5.1. As usual, let $\mathbb{Q}_q$ be the degree $[\mathbb{F}_q : \mathbb{F}_p]$ unramified extension of $\mathbb{Q}_p$. Let $C_{\mathbb{Q}_q} : y^2 = f_{\mathbb{Q}_q}(x)$ be a hyperelliptic curve in $\mathbb{Q}_p$ which reduces to C. Denote $\mathbb{C}_q := \widehat{\overline{\mathbb{Q}_q}}$ and fix an embedding $\iota : \mathbb{C}_q \hookrightarrow \mathbb{C}$, we obtain a complex abelian surface $A_{\mathbb{C}} := \mathrm{Jac}(C_{\mathbb{Q}_q})_\iota$. Assume that $A_{\mathbb{C}} \cong \mathbb{C}^2/(\mathbb{Z}^2 + \Omega\mathbb{Z}^2)$, for some lattice Ω with positive definite complex part.

Let $(a_{ij})_{0\leq i,j<2}$ be an algebric 2-theta null point of $A_{\mathbb{C}}$. The linkage between the algebraic 2-theta null points and the analytic theta functions are described by the following theorem of Carls and Lubicz:

Lemma 3.5.2 ([CL09, Lemma 2.9, page 711])**.** Let A be an abelian scheme over $\mathbb{Z}_q$, endowed with the theta structure $(A, \Theta_{2^v p}, \mathscr{L}^{2^v p})$. Also denote $Z_l = (\mathbb{Z}/l\mathbb{Z})^g$. Then there exists a $\lambda \in \mathbb{C}$, $\chi \in \hat{Z}_{2^v p}$ be a character of order 2 (that is, $\chi^2 = \mathbf{1}$ and $\delta \in Z_2$), such that for any $u \in Z_{2^v p}$, we have

$$(a_u \otimes_{\mathbb{Q}_q} \mathbb{C})_{u\in Z_{2^v p}} = \lambda\chi(u)\theta_{2^v p}\begin{bmatrix} 0 \\ u+\delta \end{bmatrix}\left(0\,,\, \frac{1}{2^v p}\Omega\right).$$

Therefore, it suffices to compute the theta functions $\theta_2\left[\begin{smallmatrix} 0 \\ u+\delta \end{smallmatrix}\right](0, \frac{1}{2}\Omega)$. The theta functions of the form $\theta_2\left[\begin{smallmatrix} u \\ v \end{smallmatrix}\right](0, \frac{1}{2}\Omega)$ can be computed from the Rosenhein form via the Thomae-Fae formula:

Theorem 3.5.3 (The Thomae formula, [Mum06, page 3.120])**.** Suppose C is a hyperelliptic curve $C : y^2 = \prod_{i=1}^{2g+1}(x - \alpha_i)$ of genus g over $\mathbb{C}$. Denote $B := \{\alpha_1, \cdots, \alpha_{2g+1}, \infty\}$ be the set of branch points. Define an abelian group $G_B := \{S \subseteq B \mid |S| \text{ is even}\}/\sim$, where $\sim$ is the smallest relation satisfying $S \sim S^c$. The group action of G_B is given by $\circ$, the "exclusive or". One readily checks that $G_B \cong (\mathbb{Z}/2\mathbb{Z})^{2g}$.

We define mappings $e_* : (\mathbb{Z}/2\mathbb{Z})^{2g} \to \{\pm 1\}$ and $e_2 : (\mathbb{Z}/2\mathbb{Z})^{2g} \times (\mathbb{Z}/2\mathbb{Z})^{2g} \to \{\pm 1\}$ as follows: Let $\xi, \zeta \in (\mathbb{Z}/2\mathbb{Z})^{2g}$, and $\xi = \left[\begin{smallmatrix} \xi_1 \\ \xi_2 \end{smallmatrix}\right]^t$ with $\xi_1, \xi_2 \in (\mathbb{Z}/2\mathbb{Z})^g$. Then $e_*(\xi) := \exp(\pi i \xi_1^t \xi_2)$; and $e_2(\xi, \zeta) := \exp(\pi i \xi^t J \zeta)$, where $J := \begin{bmatrix} 0 & \mathbf{1}_g \\ -\mathbf{1}_g & 0 \end{bmatrix}$.

Let U be a fixed subset of B of $g+1$ element such that $\infty \notin U$. For the given U, there is an isomorphism $\eta : G_B \to (\mathbb{Z}/2\mathbb{Z})^{2g}$ satisfying the following property: Denote for simplicity that $\eta(S) = \eta_S$. Then for any $S_1, S_2, T \in G_B$, $e_*(\eta_T) = (-1)^{(|T \circ U| - g - 1)/2}$ and $e_2(\eta_{S_1}, \eta_{S_2}) = (-1)^{|S_1 \cap S_2|}$.

Under the settings above, there exists a constant c, such that for any $S \in B \backslash \{\infty\}$, $|S|$ even,

$$\theta_2[\eta_S](0,\Omega)^4 = \begin{cases} 0, & \text{if } |S \circ U| \neq g+1; \\ c \cdot (-1)^{|S \cap U|} \prod_{\substack{i \in S \circ U \\ j \in B \backslash S \circ U \backslash \{\infty\}}} (\alpha_i - \alpha_j)^{-1}, & \text{if } |S \circ U| = g+1. \end{cases}$$

To apply the Thomae-Fay formula in our construction when $g = 2$, with notations the same as Theorem 3.5.3, we can choose $U = \{\alpha_1, \alpha_3, \alpha_5\}$. After U is chosen, the isomorphism η is defined as constructed in [CR15, A.2]:

$$\eta_{\{1\}} = \begin{bmatrix} (1,0) \\ (0,0) \end{bmatrix}; \quad \eta_{\{2\}} = \begin{bmatrix} (1,0) \\ (1,0) \end{bmatrix}; \quad \eta_{\{3\}} = \begin{bmatrix} (0,1) \\ (1,0) \end{bmatrix};$$
$$\eta_{\{4\}} = \begin{bmatrix} (0,1) \\ (1,1) \end{bmatrix}; \quad \eta_{\{5\}} = \begin{bmatrix} (0,0) \\ (1,1) \end{bmatrix}; \quad \eta_{\{\infty\}} = \begin{bmatrix} (0,0) \\ (0,0) \end{bmatrix}.$$

Plug in our choice of U and η and apply the Thomae-Fay formula. Note that since the algebraic theta null point is defined in the projective space, and from Lemma 3.5.2, we can rescale such that $u_{00} = 1$. And for the choice of the square root in step 1 of Proposition 3.5.1, we can choose either of them, since that only affects the choice of the basis of the period matrix Ω.

At this step, we have obtained $u_\delta = \theta_2[\begin{smallmatrix} 0 \\ \delta \end{smallmatrix}]\left(0\,,\,\frac{1}{2}\Omega\right)^2$ for each $\delta \in (\mathbb{Z}/2\mathbb{Z})^2$. We need to compute $\theta_2[\begin{smallmatrix} u \\ v \end{smallmatrix}](0,\Omega)$ for each $u, v \in (\mathbb{Z}/2\mathbb{Z})^2$. This can be done by applying the following duplication formula:

Lemma 3.5.4 (Duplication lemma, cf. [CL09, page 718]). For $u, v \in (\mathbb{Z}/2\mathbb{Z})^{2g}$,

$$\theta_2 \begin{bmatrix} v \\ u \end{bmatrix} \left(0, \frac{1}{2^{i-1}}\Omega\right)^2 = \frac{1}{2^g} \sum_{t \in (\mathbb{Z}/2\mathbb{Z})^g} (-1)^{{}^t v t} \theta_2 \begin{bmatrix} 0 \\ u+t \end{bmatrix} \left(0, \frac{1}{2^i}\Omega\right) \theta_2 \begin{bmatrix} 0 \\ t \end{bmatrix} \left(0, \frac{1}{2^i}\Omega\right).$$

Taking $g = 2, i = 1$, we get a system of equations involving a_δ and u_δ for $\delta \in (\mathbb{Z}/2\mathbb{Z})^2$ as in step 2 of Algorithm 3.5.5. As u_δ are computed now, taking a square root, we obtain a system of linear equations on a_δ, which can be readily solved. The choice of the square root in this stage does not matter as well, since this can be compensated by the character χ in Lemma 3.5.2.

Finally, from the functoriality of Jacobian, we know that starting from a hyperelliptic curve C over the finite field $\mathbb{F}_q$ of odd characteristics, we can compute the 2-theta null points over the finite field via the same equations. □

3.5.2 Computing the $2p$-theta Null Points over $\mathbb{F}_{p^r}$

For the general direction, we will be following the p-adic method proposed by Carls and Lubicz in [Car10, CL09].

3.5.2.1 Setting up the equations

Suppose that the 2-theta null points of an abelian variety A is given, denoted as $[b_{00}, b_{01}, b_{10}, b_{11}]$. One of the hardest part of a general p-adic lifting algorithm is to compute the $2p$-theta null point from the 2-theta null point. To be more precise, denote $Z(\bar{n}) = (\mathbb{Z}/n\mathbb{Z})^2$. Given the 2-theta null point $(b_u)_{u\in Z(\bar{2})}$ over a finite field, we need to find the $2p$-theta null point $(a_u)_{u\in Z(\overline{2p})}$ satisfying the requirements as in Theorem 3.2.17, and the compatibility from 2-theta null points. We summarize the condition below.

(1) (Compatibility) $a_{00} = b_{00}$, $a_{0p} = b_{01}$, $a_{p0} = b_{10}$, and $a_{pp} = b_{11}$.

(2) (Symmetry inherited from $\Theta_{\overline{nl}}$) For all $i \in Z(\overline{nl})$, $a_i = a_{-i}$.

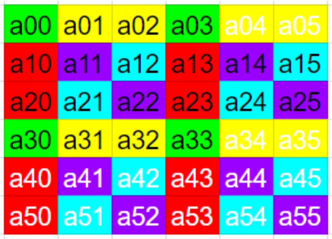
(3) (Riemann's relations) Let (v_j, w_j, x_j, y_y), $j = 1, 2$ be in $Z(\overline{nl})^4$ such that the two quadruples $(v_1 + w_1, v_1 - w_1, x_1 + y_1, x_1 - y_1)$ and $(v_2 + w_2, v_2 - w_2, x_2 + y_2, x_2 - y_2)$ only differ by permutation. Let $\chi \in Z(\bar{2})^D$. Then we have

$$\sum_{t\in Z(\bar{2})} \chi(t) a_{v_1+t} a_{w_1+t} \sum_{s\in Z(\bar{2})} \chi(s) a_{x_1+s} a_{y_1+s} = \sum_{t\in Z(\bar{2})} \chi(t) a_{v_2+t} a_{w_2+t} \sum_{s\in Z(\bar{2})} \chi(s) a_{x_2+s} a_{y_2+s}.$$

The naïve approach is to directly feed all the equations to the Gröbner basis algorithm in Magma. In other words, to compute a $2p$-theta null point, among the $(2p)^2$ entries, the symmetry relation (2) gives us $2p^2 + 2$ unique entries, and from the 4 entries coming from the 2-theta null point (as in (1)), we have $2p^2 - 2$ variables to solve from the Gröbner basis system. After plugging in the 2-theta null points in (1), the Riemann equations in (3) are polynomials of degree at most 4. This means that after eliminating $\mathbb{Z}$-linearly dependent Riemann relations, we will have at most $\binom{2p^2+2}{4} = O(p^8)$ equations to feed to the Gröbner basis system. And in the worst-case scenario, the complexity of the Gröbner basis algorithm is exponential in the number of equations, which means that it could easily get out of reach without some good improvements.

The naïve approach is infeasible even in the case $p = 3$: after removing all the perceivable redundancies, we still have 426 equations, and the Gröbner basis algorithm ran out of memory.

Carls proposed an improvement in [Car10] for computing the 6-theta null points for 3-adic lifting. The idea is to separate the 16 variables we need to solve into 4 groups (see the color codes below). And the upshot for this method is to decompose the $2p$-theta null points to $p+1$ $(2, 2p)$-theta null points lying between the 2- and $2p$-theta null points.



a00	a01	a02	a03	a04	a05
a10	a11	a12	a13	a14	a15
a20	a21	a22	a23	a24	a25
a30	a31	a32	a33	a34	a35
a40	a41	a42	a43	a44	a45
a50	a51	a52	a53	a54	a55

Figure 3.1: Grouping $2p$-theta null points. For this figure, $p = 3$. The green variable are known from the 2-theta null points, and the other 4 colors represents variables in different groups. Variables with white font colors does not need to be computed due to symmetry.

Carls explicitly wrote down 4 equations that each group of 4 variables have to satisfy (the equations are essentially the same except that the footnotes changed). However, by running through all possible Riemann relations we actually know that we have 5 $\mathbb{Z}$-linearly independent relations. Now the Gröbner basis algorithm is immediate, and resulted in 4 solutions for each of the quadruples $(a_{01}, a_{02}, a_{31}, a_{32})$, $(a_{23}, a_{20}, a_{13}, a_{10})$, $(a_{25}, a_{22}, a_{11}, a_{14})$, and $(a_{21}, a_{24}, a_{15}, a_{12})$. To combine the four groups, we simply assign them to a different solution, so there are $4! = 24$ possibilities. Note that $\mathrm{GL}(\mathbb{F}_3^2)$ acts on the 6-theta null points, whose orbit is $|\mathrm{GL}(\mathbb{F}_3^2)|/|\{\pm 1\}| = 24$, so all 24 possibilities represents an equivalent 6-null point.

And we will apply similar ideas to compute $2p$-theta null points for larget p. We were able to extend the idea on 10-theta null points ($p = 5$), this time grouping into 6 groups with 8 variables each. We end up with 30 equations, and after Gröbner basis reduction, there will be 12 solutions for each variable, yielding $2^6 \cdot 6!$ solutions. However, in contrast to the 6-theta null point case the $\mathrm{GL}(\mathbb{F}_5)$ actions only give $|\mathrm{GL}(\mathbb{F}_5)|/|\{\pm 1\}| = 240$ orbit size, so we need an extra step to test the Riemann relation to get valid 10-theta null points.

Moving one step further, for the 14-theta null points ($p = 7$), the same strategy gives 8 groups with 12 variables each, and then we have 114 $\mathbb{Z}$-linearly independent Riemann relations on each group. Unfortunately, we went out of memory while using the FGLM algorithm to compute the lexicographical Gröbner basis algorithm (step 4 in Algorithm

3.5.5). Instead, we were able to find a Gröbner basis using the Gröbner walk algorithm as in [CKM97], which was much slower than the FGLM algorithm. Currently we are still unable to find an approach for $p \geq 11$ to optimize the Gröbner basis reduction to a manageable level.

We will give our revised algorithm (Algorithm 3.5.5) on computing $2p$-theta null points in the next section, and formulate conjectural propositions which supports the validity of the algorithm.

3.5.2.2 Computing the $2p$-theta null point with our modifications

The method we applied on computing $2p$-theta null points using Gröbner bases are described below. The core idea for solving the Gröbner basis system is similar to [FLR11, Algorithm 6.3]. Our observations are applied in steps 1 and 2, with a final combination phase in step 5.

Algorithm 3.5.5 (Computing the $2p$-theta null point from 2-theta null point)**.**
Input: $(b_{ij})_{i,j\in\{0,1\}}$, the 2-theta null point of an abelian variety A.
Output: $(a_{ij})_{0\leq i,j\leq 2p}$, the $2p$-theta null point of A.

1: List all possible Riemann relations related only to a_{ij}, where $i \in \{0, 1, \cdots, p\}$, and $j = 0$ or p. Among the relations, eliminate those which are linear combinations of other Riemann relations, and pass the minimal set of Riemann relations $R \subset \mathbb{F}_q[\{a_{ij}\}_{0\leq i\leq p, j=0,p}]$ to Step 2.

2: From the set of Riemann relations R obtained from Step 1, assign the variables $(a_{00}, a_{0p}, a_{p0}, a_{pp})$ as $(b_{00}, b_{01}, b_{10}, b_{11})$. Now polynomials in R reduce to $2p-2$ variables. Among the $2p-2$ variables, take $U = \{a_{ij} \mid i \in \{2, \cdots, p-2\}, j \in \{0, p\}\}$ and $V = \{a_{10}, a_{1p}, a_{p-1,0}, a_{p-1,p}\}$.
Using Gröbner basis algorithms for `grevlex` monomial ordering (such as Faugére's F4 [Fau99] or F5 algorithm) which prioritizes in eliminating variables in U, we obtain a Gröbner basis $(G_1, <_{\texttt{grevlex}})$. G_1 is of the form $G_U \cup G_V$. Here G_U consists of $2p-6$ polynomials: for each $b_{ij} \in U$, there is one polynomial of the form $a_{ij} - p(a_{10}, a_{1p}, a_{p-1,0}, a_{p-1,p})$. And G_V contains polynomials only in V.

3: Use the FGLM algorithm as described in [FGLM93] to change the Gröbner basis $(G_V, <_{\texttt{grevlex}})$ to the lexicographical Gröbner basis $(G_2, <_{\texttt{lex}})$.

4: Use the Gröbner basis $(G_2, <_{\texttt{lex}})$ to solve all solutions of $(a_{00}, a_{0p}, a_{p0}, a_{pp})$. Let S be the set containing all quadruples of such solutions of multiplicity p^2. Then $|S| = (p^2-1)/2$. For each quadruple in S, use the partial Gröbner basis G_U from Step 3 to solve for

$a_{ij} \in U$. Note that each $(2p-6)$-uple of solution $(a_{ij})_{a_{ij}\in U}$ can be grouped into $(p-3)/2$ quadruples, and every such quadruple is contained in S.

5: The $2p$-theta null point can be written as $(b_{00}, b_{01}, b_{10}, b_{11})$ plus the $(p^2-1)/2$ disjoint quadruples of the form $(a_{2i,2j}, a_{2i,2j+p}, a_{2i+p,2j}, a_{2i+p,2j+p})$, and they are exactly the elements in S. Among the possible $(p+1)!((p-1)/2)^{p+1}$ possible assignments of the quadruples, randomly search for solutions which satisfies all the Riemann relations.

6: Return any combined $2p$-theta null point $(a_{ij})_{0\leq i,j<2p}$ which passed Step 5.

We need the following propositions for the validity of Algorithm 3.5.5:

Proposition 3.5.6. Suppose $(b_{ij})_{i,j\in\{0,1\}}$ is a 2-theta null point with respect to a marked abelian variety $(A, \Theta_{\overline{2}}, \mathscr{L})$ (see Definition 3.2.14 for details). Let (δ_1, δ_2) satisfy $2 \mid \delta_1 \mid \delta_2$. Then there exists a (δ_1, δ_2)-theta null point $(a_{i'j'})_{0\leq i'<\delta_1}^{0\leq j'<\delta_2}$ corresponding to $(A, \Theta_{\overline{(\delta_1,\delta_2)}}, \mathscr{L}')$ and for each $i, j \in \{0, 1\}$, $b_{ij} = a_{i\delta_1/2, j\delta_2/2}$.

We need the case $(2p, 2p)$ for the first half of step 2 and the case $(2, 2p)$ for the second half of step 2. Probably need to connect the algebraic theta theory to the analytic side.

Proposition 3.5.7. Denote $W = U \cup V$, where $U = \{a_{ij} \mid i \in \{2, \cdots, p-2\}, j \in \{0, p\}\}$ and $V = \{a_{10}, a_{1p}, a_{p-1,0}, a_{p-1,p}\}$, as in step 2 of Algorithm 3.5.5. The values of $\{a_{00}, a_{0p}, a_{p0}, a_{pp}\}$ and the equations in Theorems 3.2.16 involving variables in W and $\{a_{00}, a_{0p}, a_{p0}, a_{pp}\}$ defines a dimension 0 scheme in the affine space $\mathbb{A}_W$, where $(p^2-1)/2$ points have multiplicity p^2, 1 point has multiplicity p and 1 point has multiplicity 1.

And for the feasibility of step 5 in Algorithm 3.5.5, we need the following proposition.

Proposition 3.5.8. Suppose S is the set consisting of quadruples of V-coordinates of points in the 0-dimensional scheme as in Proposition 3.5.7 which has multiplicity p^2 (so from Proposition 3.5.7, $|S| = (p^2-1)/2$). Then there are $2p$-theta null points $(a_{ij})_{0\leq i,j<2p}$ satisfying the symmetric and Riemann relations, as well as the following properties:

(1) $(a_{00}, a_{0p}, a_{p0}, a_{pp}) = (b_{00}, b_{01}, b_{10}, b_{11})$, and

(2) For each pair $(i, j) \neq (0, 0)$ satisfying $0 \leq i, j < p$, the quadruple $(a_{ij}, a_{i,j+p}, a_{i+p,j}, a_{i+p,j+p})$ is in S, and distince (i, j) corresponds to distinct quadruples.

These propositions are not yet solved. We believe that generalizing the $\mathrm{Aut}_{\mathbb{G}_m} H_{\overline{2p}}$-actions as in [FLR11] is the key to these propositions and will attempt to finish this up as future work. See [FLR11, Section 5] for more details.

3.5.3 Computing the $2p$-theta Null Points over $\mathbb{Q}_{p^r}$

At this stage, we have computed the $2p$-theta null points $(a_{ij})_{0\le i,j<2p}$ of an abelian variety A defined over $\mathbb{F}_{p^r}$. Then, from [Mes72], there is a canonical lift $\tilde{A}$ defined over the Witt ring $W(\mathbb{F}_{p^r}) \cong \mathbb{Z}_{p^r}$, the degree r unramified extension of $\mathbb{Z}_p$. Now, it suffices to compute the $2p$-theta null points $(\tilde{a}_{ij})_{0\le i,j<2p}$ of $\tilde{A}$.

For $(\tilde{a}_{ij})_{0\le i,j<2p}$ to be a theta null point of the canonical lifting, we will need $(\tilde{a}_{ij})_{0\le i,j<2p}$ to reduce to $(a_{ij})_{0\le i,j<2p}$, plus they have to satisfy the following theorem by Carls and Lubicz:

Theorem 3.5.9 ([CL09, Theorem 2.1]). Suppose x, y, v, w are elements in $\mathbb{Z}/2p\mathbb{Z}$, and the sets $\{x+y, x-y\}$ and $\{v+pw, v-pw\}$ are the same sets and both contained in $\mathbb{Z}/p\mathbb{Z}$. Denote by σ the p-th power Frobenius on $\mathbb{Z}_{p^r}$. Then there is an element $\omega \in \mathbb{Z}_{p^r}^*$ such that

$$\sum_{z\in\mathbb{Z}/2\mathbb{Z}} \tilde{a}_{x+z}\tilde{a}_{y+z} = \omega \sum_{u\in\mathbb{Z}/2p\mathbb{Z}} \tilde{a}_{v+pu}\tilde{a}_{w+u}^{\sigma^2}.$$

The correspondence relation in Theorem 3.5.9 together with the symmetric and Riemann relations in Theorem 3.2.17 determines the $2p$-theta null point of $(\tilde{a}_{ij})_{0\le i,j<2p}$.

While theoretically it is possible to use Theorems 3.5.9 and 3.2.16 to determine $(\tilde{a}_{ij})_{0\le i,j<2p}$ in $\mathbb{Q}_{p^r}$, computationally we can only find them up to a precision N, equivalently, find the reductions of $(\tilde{a}_{ij})_{0\le i,j<2p}$ in $\mathbb{Z}_{p^r}/p^N\mathbb{Z}_{p^r}$. The requirement on N is that it should be large enough to recover the Igusa class polynomials, so it is determined by the size of the coefficients and the degree of the Igusa class polynomials, and the errors introduced by the LLL algorithm in Section 5. We should discuss the choice of N in Section 3.8.4, which is affected by the effectiveness of LLL algorithm in finding the shortest vectors and the size of coefficients of the Igusa class polynomial. In short, $N \in \tilde{O}(D_0^{5/2}D_1^{3/2})$.

as one can proceed by the following algorithm.

Algorithm 3.5.10 ([CL09, Section 3.2], canonical lifting of an abelian surface over $\mathbb{F}_{p^r}$).
Input: The $2p$-theta null point $(a_{ij})_{0\le i,j<2p} \in \mathrm{Mat}_{2p}(\mathbb{F}_{p^r})$ of an abelian surface, a prescribed p^r-adic precision N.
Output: The $2p$-theta null point $(\tilde{a}_{ij})_{0\le i,j<2p} \in \mathrm{Mat}_{2p}(\mathbb{Q}_{p^r}/p^N\mathbb{Q}_{p^r})$ of the canonical lifting, up to precision N.

```
1: Pick up a set of $2p^2 - 2$ non-singular equations from the Riemann relations.
2: Pick up a set of 3 non-singular equations from the correspondence relations.
3: After projectification, the $2p^2 + 1$ equations chosen from steps 1 and 2 defines a system
```

of equation $\Phi : \mathbb{Z}_{p^r}^{2p^2+1} \times \mathbb{Z}_{p^r}^{2p^2+1} \to \mathbb{Z}_{p^r}^{2p^2+1}$, written as

$$\Phi(x,y) = (f_1(x,y), \cdots, f_{2p^2+1}(x,y)).$$

```
4: Solve the equation $\Phi(A, A^{\sigma^2}) \equiv 0 \pmod{p^N}$, where $\sigma \in \mathrm{Gal}(\mathbb{Q}_{p^r}/\mathbb{Q}_p)$ is the Frobenius in the totally ramified extension.
5: Return the affine version of $A$.
```

Steps 1 and 2 only involve selecting equations with coefficients in $\mathbb{Z}$, and the time complexity is negligible. For the dominating step 3, it is where the canonical lifting takes place. Using the relation in step 3 to find the canonical lifting is indeed the higher dimensional analog of the Newton method or the Hensel lifting.

Suppose we know at some stage the solution A in step 3 up to precision $m/2$. This means that we have $\Phi(A, A^{\sigma^2}) \equiv 0 \pmod{p^{m/2}}$. To enhance the solution to precision m, we then need to solve $\Delta \in \mathbb{Q}_{p^d}/p^{m/2}\mathbb{Q}_{p^d}$ such that

$$\Phi\left(A + p^{m/2}\Delta, A^{\sigma^2} + p^{m/2}\Delta^{\sigma^2}\right) \equiv 0 \pmod{p^m}.$$

Let D_X and D_Y be $(2p^2+1) \times (2p^2+1)$ matrices whose ij-th entry is given by the x_i- and y_j-parial derivatives of f_i, respectively, Then from the Taylor's expansion, we have

$$0 \equiv \Phi(A, A^{\sigma^2}) + p^{m/2} D_X(A, A^{\sigma^2})\Delta + p^{m/2} D_Y(A, A^{\sigma/2})\Delta^{\sigma^2} \pmod{p^m}.$$

So, in each iteration of the canonical lifting, it suffices to solve the Artin-Schrier equation of the form $\Delta^{\sigma^2} = A\Delta + B$, where $A, B \in \mathrm{Mat}_{(2p^2+1)\times(2p^2+1)}(\mathbb{Q}_{p^d}/p^{m/2}\mathbb{Q}_{p^d})$.

To solve the Artin-Schreier equation of the form $\Delta^{\sigma^2} = A\Delta + B$, let $e = r/2$ or r be the minimal exponent such that $(\Delta^{\sigma^2})^e = \Delta$. Then one can iterate the Artin-Schrier equation: for example, $\Delta^{\sigma^4} = (A\Delta + B)^{\sigma^2} = A^{\sigma^2}\Delta^{\sigma^2} + B^{\sigma^2} = A^{\sigma^2}(A\Delta + B) + B^{\sigma^2}$. From that, we can get a linear system involving only $\Delta^{\sigma^{2e}} = \Delta$. [LL06, Theorem 2] gave condition on when this linear system is non-singular, which is the general case. In case we end up with a singular linear system, we will have to randomly choose equations in steps 1 and 2 again.

3.6 Recovering the Igusa Class Polynomials from the Canonical Lift

At this stage, we have the theta coefficients of an abelian variety $\tilde{A}$ defined over $\mathbb{Q}_{p^r}$ which has CM by $\mathcal{O}_K$, and the theta coefficients has a sufficient precision N. In this step, we will recover factors of the Igusa class polynomials for the quartic CM field K. This can be achieved by the following steps:

(1) Compute the Igusa invariants $(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$ of the abelian variety with the $2p$-theta null points $(\tilde{a}_{ij})_{0 \leq i,j < 2p}$, up to precision N

(2) Use the LLL algorithm to compute the minimal polynomials $\{\widetilde{h_{K,1}}, \widetilde{h_{K,2}}, \widetilde{h_{K,3}}\}$ of each Igusa coefficient. See Algorithm 3.6.1.

(3) Check if the degrees of $\widetilde{h_{K,1}}$ are as expected. If not, find another abelian variety over $\mathbb{F}_{p^r}$ and go through the canonical lifting.

For step (1), starting from the $2p$-theta null points of the canonical lifting $\tilde{A}$, we can pick $(\tilde{a}_{00}, \tilde{a}_{0p}, \tilde{a}_{p0}, \tilde{a}_{pp})$ to obtain the 2-theta null points. Then, by reverting the steps in Proposition 3.5.1, we can get the Rosenhein invariants of a hyperelliptic curve $\tilde{C}$ such that $\tilde{A} \cong \mathrm{Jac}(\tilde{C})$. Then we can obtain the Igusa invariants $(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$ from the equation of $\tilde{C}$.

For step (2), we can apply LLL-algorithm to find the minimal polynomial of the Igusa invariants, as stated in Algorithm 3.6.1.

Algorithm 3.6.1 (Recovering Igusa class polynomials from Igusa invariants, [GHK+06, Section 4.2])**.**
Input: A local field extension $\mathbb{Z}_{p^d}/\mathbb{Z}_p$, with $\mathbb{Z}_p$-basis $\{w_0 = 1, w_1, \cdots, w_{d-1}\}$, an Igusa invariant $\iota \in \mathbb{Z}_{p^d}/p^N\mathbb{Z}_{p^d}$ with precision N, the expected degree s of the Igusa class polynomial
Output: The minimal polynomial $f(x) = \sum_{i=0}^{s} m_i x^i$ of ι

```
Compute the $(s+d+1) \times d$ matrix $M = \begin{bmatrix} A \\ p^N I_d \end{bmatrix}$, where $A = (a_{ij}) \in \mathrm{Mat}(\mathbb{Z}, s+1, d)$,
    where $a_{ij}$ is in $[0, p^N)$ and satisfies
        $\iota^0 = a_{0,0}w_0 + \cdots + a_{0,d-1}w_{d-1},$
        $\iota^1 = a_{1,0}w_0 + \cdots + a_{1,d-1}w_{d-1},$
        $\vdots$
        $\iota^s = a_{s,0}w_0 + \cdots + a_{s,d-1}w_{d-1},$
    when reduced to $\mathbb{Z}_p/p^N\mathbb{Z}_p$.
Let $\Lambda$ be the left kernel of $M$. Use LLL to compute the shortest vector of $\Lambda$. Suppose it
    is $[m_0\ m_1\ \cdots\ m_s - \epsilon_0\ \cdots\ -\epsilon_{d-1}]$.
Return $f(x) = \sum_{i=0}^{s} m_i x^i$.
```

Note that while the approach is seen in [GHK$^+$06], a similar minimal polynomial algorithm in the complex analytic setting can be found in [Str14]. The precision N is the same as the one used in Section 3.5.3, which is of size $\tilde{O}(D_0^{5/2}D_1^{3/2})$. We will establish the validity of the algorithm as well as the estimation of N in Section 3.8.4.

3.7 Examples

We will present two examples, which finds the Igusa class polynomials of the cyclic number field $\mathbb{Q}(\sqrt{-2+\sqrt{2}})$, including a canonical lift over $\overline{\mathbb{F}_7}$; as well as the non-Galois number field $\mathbb{Q}(\sqrt{-30+\sqrt{96}})$, including a canonical lift over $\overline{\mathbb{F}_5}$. The examples are computed by Magma v2.25 on a Intel(R) Core(TM) i9-9980HK CPU with 1.3GB RAM.

3.7.1 Example 1: $\mathbb{Q}\left(\sqrt{-2+\sqrt{2}}\right)$

Todo: When finished, break down the bullet points...

1. We denote $K = \mathbb{Q}\left(\sqrt{-2+\sqrt{2}}\right)$. By testing small primes, we can immediately see that $p = 7$ splits completely in K, which implies that there exists hyperelliptic Jacobians defined over $\mathbb{F}_7$ with CM by $\mathcal{O}_K$.

2. We can compute that for a hyperelliptic curve $C/\mathbb{F}_7$, if we denote by n_C and n_J the number of points of C and $\mathrm{Jac}(C)$ over $\mathbb{F}_7$, then $\mathrm{Jac}(C)$ has CM by $\mathcal{O}_K$ only when $(n_C, n_J) = (4, 28)$ or $(12, 92)$.

3. We run through all possible Igusa invariants $(i_1, i_2, i_3) \in \mathbb{F}_7^3$, we can see that

$$C : y^2 = 3x^6 + 3x^5 + 5x^4 + x^3 + 6x^2 + 2x + 6$$

   has the right orders. And then we pass to the endomorphism algorithm by [Koh], we verify that $\mathrm{Jac}(C)$ does have CM by the maximal order $\mathcal{O}_K$.

4. Then, we need to compute the canonical lift of C. The first step is to compute its 14-theta null point. We need to start by computing the Rosenhein form of C and deduct the 2-theta null point from there. It turns out that, over the field extension $\mathbb{F}_7[\tau]/(\tau^2+6\tau+3) \cong \mathbb{F}_{7^2}$, C is isomorphic to $C' : y^2 = x(x-1)(x-3)(x-5)(x-\tau^6)$. From the explicit formulas in Proposition 3.5.1, we get the following 2-theta null point, defined in $\mathbb{F}_{7^2}[\sigma]/(\sigma^2+\tau^9) \cong \mathbb{F}_{7^4}$:

$$\begin{bmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \end{bmatrix} = \begin{bmatrix} \tau^{38}\sigma + \tau^{27} & \tau^{20}\sigma + \tau^{19} \\ \tau^{10}\sigma + 3 & \tau^{35}\sigma + \tau^{18} \end{bmatrix}.$$

5. Now we are at the most computationally heavy part–computing the 14-theta null point $(a_{ij})_{0\leq i,j<14}$. From the symmetric relation $a_{ij} = a_{-i,-j}$, essentially we still need to determine the value of 100 coordinates (including those inherited from the 2-theta null points).

   To trim down the computational complexity of the Gröbner basis algorithm, as discussed in Algorithm 3.5.5, we shall regroup the $2p^2 - 2$ variables into $p+1$ groups, each consisting of variables which forms a $(2, 2p)$-theta structure and satisfies the same set of Riemann relation equations. For example, one of the groups consists of the 12 variables $(a_{ij})_{1\leq i\leq p-1}^{j=0,p}$. As in step 2 of Algorithm 3.5.5, we applied the F4 algorithm to compute the first Gröbner basis in the `grevlex` order, and to boost the computation of Gröbner basis, we priortize on eliminating the 8 variables $U = \{a_{20}, a_{27}, \cdots, a_{50}, a_{57}\}$ and get equations on $V = \{a_{10}, a_{17}, a_{60}, a_{67}\}$. The first Gröbner basis computation took about 26 seconds to finish, and returned a basis with 695 polynomials: 8 of them are only of degree 1 with respect to the variables in U, and the other 687 of them are solely on variables in V.

6. With the equations on V, to solve them, we will compute the second Gröbner basis, now on the lexicographical `lex` order. Typically we use the FGLM algorithm [FGLM93], however, as it slightly went beyond the memory limit with the Magma implementation, we have to take the Gröbner walk algorithm. The second Gröbner basis computation took about 70 minutes on our platform. On the Gröbner basis, the last polynomial is univariate on b_{67} of degree 1184, consisting of $(p^2-1)/2 = 24$ roots of multiplicity $p^2 = 49$, 1 root of multiplicity $p = 7$ and 1 trivial root 0 of multiplicity 1.

   Solving for the 24 roots, we can recover 24 solution sets for a_{67}. All the roots will be in the field $\mathbb{F}_{7^4}[\nu]/(\nu^3 + \tau^{37}\sigma + \tau^{12}) \cong \mathbb{F}_{7^{12}}$. And from this we can sequentially solve the equations in the Gröbner basis, and apply all the $\mathrm{GL}_2(\mathbb{F}_7)$ actions, we see that the 24 quadruples of variables

$$\begin{aligned}\{q_1, \cdots, q_{24}\} = \{&(a_{10}, a_{17}, a_{60}, a_{67}), (a_{30}, a_{37}, a_{40}, a_{47}), (a_{50}, a_{57}, a_{20}, a_{27}), (a_{76}, a_{71}, a_{06}, a_{01}), (a_{74}, a_{73}, a_{04}, a_{03}),\\ &(a_{72}, a_{75}, a_{02}, a_{05}), (a_{12}, a_{19}, a_{6,12}, a_{65}), (a_{36}, a_{3,13}, a_{48}, a_{41}), (a_{5,10}, a_{53}, a_{24}, a_{2,11}), (a_{14}, a_{1,11}, a_{6,10}, a_{63}),\\ &(a_{3,12}, a_{35}, a_{42}, a_{49}), (a_{56}, a_{5,13}, a_{28}, a_{21}), (a_{16}, a_{1,13}, a_{68}, a_{61}), (a_{34}, a_{3,11}, a_{4,10}, a_{43}), (a_{52}, a_{59}, a_{2,12}, a_{25}),\\ &(a_{18}, a_{11}, a_{66}, a_{6,13}), (a_{3,10}, a_{33}, a_{44}, a_{4,11}), (a_{5,12}, a_{55}, a_{22}, a_{29}), (a_{1,10}, a_{13}, a_{64}, a_{6,11}), (a_{32}, a_{39}, a_{4,12}, a_{45}),\\ &(a_{58}, a_{51}, a_{26}, a_{2,13}), (a_{1,12}, a_{15}, a_{62}, a_{69}), (a_{38}, a_{31}, a_{46}, a_{4,13}), (a_{54}, a_{5,11}, a_{2,10}, a_{23})\}\end{aligned}$$

   in a 14-theta null point will be bijectively matched to the set of values $\{\theta_1, \cdots, \theta_{24}\}$, which are:

$$
\begin{aligned}
&\{(\tau^{47}\sigma+4, \tau^{41}\sigma, \tau^{39}\sigma+\tau^{23}, \tau^{30}\sigma+\tau^{27}),\\
&((\tau^{30}\sigma+\tau^{26})\nu^2+(\tau^{22}\sigma+\tau^5)\nu+\tau^{10}\sigma+5, (6\sigma+\tau^6)\nu^2+(5\sigma+\tau^7)\nu+\tau^{31}\sigma+\tau^{41}, (\tau^{21}\sigma+\tau^6)\nu^2+(\tau^{22}\sigma+\tau^5)\nu+\tau^{44}\sigma+\tau^{39}, \tau^3\nu^2+(\tau^{26}\sigma+\tau^{43})\nu+\tau^{29}\sigma+\tau^{37}),\\
&(\tau^3\sigma+\tau^{15}, \tau^{27}\sigma+1, 4\sigma+\tau^{23}, \tau^{35}\sigma+\tau^{29}),\\
&(\tau^{19}\sigma+\tau^{20}, \tau^{13}, \tau^4\sigma+\tau^4, \tau^{38}\sigma+\tau^{26}),\\
&((\tau^{17}\sigma+\tau^{18})\nu^2+(\tau^4\sigma+\tau^{25})\nu+\tau^{43}\sigma+\tau^{10}, (\tau^{33}\sigma+\tau^{28})\nu^2+(\tau^{30}\sigma+\tau^{31})\nu+\tau^{46}\sigma+\tau^{46}, (\tau^5\sigma+\tau^9)\nu^2+(\tau^6\sigma+\tau^{43})\nu+\tau^5\sigma+\tau^{46}, (4\sigma+\tau^{23})\nu^2+(\tau^{22}\sigma+\tau^{12})\nu+\tau^{37}\sigma+\tau^{38}),\\
&((\tau\sigma+\tau^4)\nu^2+(\tau^{10}\sigma+3)\nu+\tau^{34}\sigma+\tau^{39}, (\tau^{18}\sigma+\tau^7)\nu^2+(\tau^{39}\sigma+\tau^{22})\nu+2\sigma+\tau^{19}, \tau^{46}\sigma\nu^2+(\tau^{28}\sigma+\tau^{35})\nu+\tau^{27}\sigma+\tau^{36}, (\tau^{46}\sigma+\tau^{46})\nu^2+(\tau^{15}\sigma+\tau^{46})\nu+\tau^{15}\sigma+\tau^5),\\
&((\tau^2\sigma+\tau^{43})\nu^2+(\tau^3\sigma+\tau^4)\nu+4\sigma+6, (\tau^{31}\sigma+\tau^{20})\nu^2+(\tau^{13}\sigma+\tau^4)\nu+3\sigma+2, (\tau^{31}\sigma+\tau^{37})\nu^2+(\tau^{19}\sigma+\tau^9)\nu+\tau^{20}\sigma+\tau^{38}, (\tau^{11}\sigma+\tau^{11})\nu^2+(\tau^{41}\sigma+\tau^{26})\nu+\tau^{10}\sigma+\tau^{20}),\\
&((\tau^{41}\sigma+\tau^{37})\nu^2+(\tau^{33}\sigma+\tau^4)\nu+\sigma+\tau^9, (\tau^{25}\sigma+\tau^{13})\nu^2+(\tau^{30}\sigma+\tau^{25})\nu+\tau^{19}\sigma+\tau^{39}, (4\sigma+\tau^{17})\nu^2+(\tau^{12}\sigma+\tau^{21})\nu+\tau^7\sigma+\tau^{12}, (\tau^7\sigma+\tau^{26})\nu^2+(\tau^{25}\sigma+\tau^9)\nu+5\sigma+\tau^9),\\
&((\tau^{42}\sigma+\tau^{33})\nu^2+(\tau^{21}\sigma+\tau^{38})\nu+\tau^{18}\sigma+\tau^{43}, (\tau^3\sigma+\tau^{25})\nu^2+(\tau^{39}\sigma+\tau^{37})\nu+\tau^{14}\sigma+\tau^{38}, (\tau^{19}\sigma+\tau^{14})\nu^2+(\tau^{11}\sigma+\tau^{26})\nu+\tau^{22}\sigma+\tau^{45}, (\tau^{38}\sigma+\tau^{18})\nu^2+(\tau^{33}\sigma+5)\nu+\tau^{17}\sigma+\tau^{45}),\\
&((\tau^{17}\sigma+\tau^{20})\nu^2+(\tau^{42}\sigma+5)\nu+\tau^{34}\sigma+\tau^{39}, (\tau^{34}\sigma+\tau^{23})\nu^2+(\tau^{23}\sigma+\tau^6)\nu+2\sigma+\tau^{19}, \tau^{14}\sigma\nu^2+(\tau^{12}\sigma+\tau^{19})\nu+\tau^{27}\sigma+\tau^{36}, (\tau^{14}\sigma+\tau^{14})\nu^2+(\tau^{47}\sigma+\tau^{30})\nu+\tau^{15}\sigma+\tau^5),\\
&((\tau^{25}\sigma+\tau^{21})\nu^2+(\tau\sigma+\tau^{20})\nu+\sigma+\tau^9, (\tau^9\sigma+\tau^{45})\nu^2+(\tau^{46}\sigma+\tau^{41})\nu+\tau^{19}\sigma+\tau^{39}, (2\sigma+\tau)\nu^2+(\tau^{28}\sigma+\tau^{37})\nu+\tau^7\sigma+\tau^{12}, (\tau^{39}\sigma+\tau^{10})\nu^2+(\tau^{41}\sigma+\tau^{25})\nu+5\sigma+\tau^9),\\
&((\tau^{45}\sigma+\tau^{19})\nu^2+(\tau^6\sigma+\tau^{38})\nu+\tau^{43}, (\tau^{10}\sigma+\tau^{39})\nu^2+(\tau^{45}\sigma+\tau^9)\nu+\tau^{41}\sigma+\tau^6, (3\sigma+\tau^{36})\nu^2+(\tau^{35}\sigma+\tau^{30})\nu+\tau^{44}\sigma+4, (\tau^{19}\sigma+\tau^3)\nu^2+(\tau^{15}\sigma+\tau^{11})\nu+\tau^{35}\sigma+\tau^3),\\
&((\tau^{33}\sigma+\tau^{34})\nu^2+(\tau^{36}\sigma+\tau^9)\nu+\tau^{43}\sigma+\tau^{10}, (\tau\sigma+\tau^{44})\nu^2+(\tau^{14}\sigma+\tau^{15})\nu+\tau^{46}\sigma+\tau^{46}, (\tau^{21}\sigma+\tau^{25})\nu^2+(\tau^{38}\sigma+\tau^{27})\nu+\tau^5\sigma+\tau^{46}, (\sigma+\tau^{39})\nu^2+(\tau^6\sigma+\tau^{44})\nu+\tau^{37}\sigma+\tau^3 8),\\
&((\tau^{29}\sigma+\tau^3)\nu^2+(\tau^{22}\sigma+\tau^6)\nu+\tau^{43}, (\tau^{42}\sigma+\tau^{23})\nu^2+(\tau^{13}\sigma+\tau^{25})\nu+\tau^{41}\sigma+\tau^6, (5\sigma+\tau^{20})\nu^2+(\tau^3\sigma+\tau^{46})\nu+\tau^{44}\sigma+4, (\tau^3\sigma+\tau^{35})\nu^2+(\tau^{31}\sigma+\tau^{27})\nu+\tau^{35}\sigma+\tau^3),\\
&((\tau\sigma+\tau^2)\nu^2+(\tau^{20}\sigma+\tau^{41})\nu+\tau^{43}\sigma+\tau^{10}, (\tau^{17}\sigma+\tau^{12})\nu^2+(\tau^{46}\sigma+\tau^{47})\nu+\tau^{46}\sigma+\tau^{46}, (\tau^{37}\sigma+\tau^{41})\nu^2+(\tau^{22}\sigma+\tau^{11})\nu+\tau^5\sigma+\tau^{46}, (2\sigma+\tau^7)\nu^2+(\tau^{38}\sigma+\tau^{28})\nu+\tau^{37}\sigma+\tau^{38}),\\
&((\tau^{33}\sigma+\tau^{36})\nu^2+(\tau^{26}\sigma+6)\nu+\tau^{34}\sigma+\tau^{39}, (\tau^2\sigma+\tau^{39})\nu^2+(\tau^7\sigma+\tau^{38})\nu+2\sigma+\tau^{19}, \tau^{30}\sigma\nu^2+(\tau^{44}\sigma+\tau^3)\nu+\tau^{27}\sigma+\tau^{36}, (\tau^{30}\sigma+\tau^{30})\nu^2+(\tau^{31}\sigma+\tau^{14})\nu+\tau^{15}\sigma+\tau^5),\\
&((\tau^{13}\sigma+\tau^{35})\nu^2+(\tau^{38}\sigma+\tau^{22})\nu+\tau^{43}, (\tau^{26}\sigma+\tau^7)\nu^2+(\tau^{29}\sigma+\tau^{41})\nu+\tau^{41}\sigma+\tau^6, (6\sigma+\tau^4)\nu^2+(\tau^{19}\sigma+\tau^{14})\nu+\tau^{44}\sigma+4, (\tau^{35}\sigma+\tau^{19})\nu^2+(\tau^{47}\sigma+\tau^{43})\nu+\tau^{35}\sigma+\tau^3),\\
&((\tau^{34}\sigma+\tau^{27})\nu^2+(\tau^{19}\sigma+\tau^{20})\nu+4\sigma+6, (\tau^{15}\sigma+\tau^4)\nu^2+(\tau^{29}\sigma+\tau^{20})\nu+3\sigma+2, (\tau^{15}\sigma+\tau^{21})\nu^2+(\tau^{35}\sigma+\tau^{25})\nu+\tau^{20}\sigma+\tau^{38}, (\tau^{43}\sigma+\tau^{43})\nu^2+(\tau^9\sigma+\tau^{42})\nu+\tau^{10}\sigma+\tau^{20}),\\
&((\tau^9\sigma+\tau^5)\nu^2+(\tau^{17}\sigma+\tau^{36})\nu+\sigma+\tau^9, (\tau^{41}\sigma+\tau^{29})\nu^2+(\tau^{14}\sigma+\tau^9)\nu+\tau^{19}\sigma+\tau^{39}, (\sigma+\tau^{33})\nu^2+(\tau^{44}\sigma+\tau^5)\nu+\tau^7\sigma+\tau^{12}, (\tau^{23}\sigma+\tau^{42})\nu^2+(\tau^9\sigma+\tau^{41})\nu+5\sigma+\tau^9),\\
&((\tau^{18}\sigma+\tau^{11})\nu^2+(\tau^{35}\sigma+\tau^{36})\nu+4\sigma+6, (\tau^{47}\sigma+\tau^{36})\nu^2+(\tau^{45}\sigma+\tau^{36})\nu+3\sigma+2, (\tau^{47}\sigma+\tau^5)\nu^2+(\tau^3\sigma+\tau^{41})\nu+\tau^{20}\sigma+\tau^{38}, (\tau^{27}\sigma+\tau^{27})\nu^2+(\tau^{25}\sigma+\tau^{10})\nu+\tau^{10}\sigma+\tau^{20}),\\
&((\tau^{26}\sigma+\tau^{17})\nu^2+(\tau^{37}\sigma+\tau^6)\nu+\tau^{18}\sigma+\tau^{43}, (\tau^{35}\sigma+\tau^9)\nu^2+(\tau^7\sigma+\tau^5)\nu+\tau^{14}\sigma+\tau^{38}, (\tau^3\sigma+\tau^{46})\nu^2+(\tau^{27}\sigma+\tau^{42})\nu+\tau^{22}\sigma+\tau^{45}, (\tau^{22}\sigma+\tau^2)\nu^2+(\tau\sigma+3)\nu+\tau^{17}\sigma+\tau^{45}),\\
&((\tau^{10}\sigma+\tau)\nu^2+(\tau^5\sigma+\tau^{22})\nu+\tau^{18}\sigma+\tau^{43}, (\tau^{19}\sigma+\tau^{41})\nu^2+(\tau^{23}\sigma+\tau^{21})\nu+\tau^{14}\sigma+\tau^{38}, (\tau^{35}\sigma+\tau^{30})\nu^2+(\tau^{43}\sigma+\tau^{10})\nu+\tau^{22}\sigma+\tau^{45}, (\tau^6\sigma+\tau^{34})\nu^2+(\tau^{17}\sigma+6)\nu+\tau^{17}\sigma+\tau^{45}),\\
&((\tau^{14}\sigma+\tau^{10})\nu^2+(\tau^{38}\sigma+\tau^{21})\nu+\tau^{10}\sigma+5, (3\sigma+\tau^{38})\nu^2+(3\sigma+\tau^{23})\nu+\tau^{31}\sigma+\tau^{41}, (\tau^5\sigma+\tau^{38})\nu^2+(\tau^{38}\sigma+\tau^{21})\nu+\tau^{44}\sigma+\tau^{39}, \tau^{35}\nu^2+(\tau^{42}\sigma+\tau^{11})\nu+\tau^{29}\sigma+\tau^{37}),\\
&((\tau^{46}\sigma+\tau^{42})\nu^2+(\tau^6\sigma+\tau^{37})\nu+\tau^{10}\sigma+5, (5\sigma+\tau^{22})\nu^2+(6\sigma+\tau^{39})\nu+\tau^{31}\sigma+\tau^{41}, (\tau^{37}\sigma+\tau^{22})\nu^2+(\tau^6\sigma+\tau^{37})\nu+\tau^{44}\sigma+\tau^{39}, \tau^{19}\nu^2+(\tau^{10}\sigma+\tau^{27})\nu+\tau^{29}\sigma+\tau^{37})\}.
\end{aligned}
$$

We took random testing to find a correct match. It took us about 1.08 million trials until a valid 14-theta null point is found, which took about 5 minutes. In the correct combination, $(q_1, \cdots, q_{24})$ is mapped to

$$(\theta_{24}, \theta_5, \theta_8, \theta_6, \theta_{14}, \theta_7, \theta_{13}, \theta_{19}, \theta_{23}, \theta_{20}, \theta_{10}, \theta_{12}, \theta_9, \theta_{21}, \theta_{22}, \theta_4, \theta_3, \theta_1, \theta_{17}, \theta_{18}, \theta_{16}, \theta_2, \theta_{15}, \theta_{11}).$$

respectively. And we can assemble the 14-theta null point from the mapping.

7. After the 14-theta null point over $\mathbb{F}_{7^{12}}$ is computed, we need to pass to Algorithm 3.5.10 to compute the lifted theta null point over $\mathbb{Q}_7[t]/(t^{12}+2t^8+5t^7+3t^6+2t^5+4t^4+5t^2+3) \cong Q_{7^{12}}$, the degree 12 unramified extension of $\mathbb{Q}_7$. As described in Algorithm 3.5.10, we will need to find 96 Riemann equations from Theorem 3.2.17 and 3 correspondence equations from Theorem 3.5.9 which forms a non-singular system of Artin-Schreier equation in the 99-dimensional projective space over $\mathbb{Q}_{7^{12}}$. After testing, we know that $N = 32$ 7-adic digits precision is sufficient to recover the Igusa polynomials after the LLL-reduction in Algorithm 3.6.1. The canonical lift step took about 122 seconds. And after sampling, the lifted 2-theta null point in

$\mathbb{Q}_{7^{12}}/7^{32}\mathbb{Q}_{7^{12}}$ is $(\tilde{a}_{00}, \tilde{a}_{07}, \tilde{a}_{70}, \tilde{a}_{77})$, where

$$\begin{aligned}
\tilde{a}_{00} &= 1,\\
\tilde{a}_{07} &= 4302099632335167393438276 84t^{11} - 2630389416875846992003690 49t^{10} + 7351142214024374253803014 3t^{9}\\
&\quad - 482917741485237891852802839t^{8} + 653621877504103456401906 40t^{7} + 387944182120549706869198782t^{6}\\
&\quad + 537884498732469939141763739t^{5} + 129600731917656931268405 17t^{4} + 279311473845499276425640053t^{3}\\
&\quad + 398664681377083153281231992t^{2} + 458708656158401987705625398t + 126708230087604433229111745,\\
\tilde{a}_{70} &= -200019157013160998614803282t^{11} + 255563578482595331287629 71t^{10} - 508706914512636967017762957t^{9}\\
&\quad - 241002713401785217617038092t^{8} - 512802758811656154351955312t^{7} - 980602477163204460970336 63t^{6}\\
&\quad - 140576647483119572847929766t^{5} - 260688413071446174461808 61t^{4} - 472922291241081750675627331t^{3}\\
&\quad + 161291271265340183894296957t^{2} - 318424673281497759260444088t - 531872481349200705504981357,\\
\tilde{a}_{77} &= -169532528779527646088988256t^{11} + 524082264939474138643949 52t^{10} - 291558267657077035223235 1t^{9}\\
&\quad + 149253476431052829230311408t^{8} - 522657937084187089539633274t^{7} + 146885636140666855930472454t^{6}\\
&\quad - 484807571865515135169724303t^{5} + 211935525808214294590925139t^{4} - 409632555841931027499097093t^{3}\\
&\quad + 355771127208983547553200251t^{2} - 418526467847341521551028766t - 474600739522272172687113429.
\end{aligned}$$

8. From the lifted 2-theta null point, we can recover in the order the Rosenhein invariants, the lifted hyperelliptic curve $\tilde{C}$, and the Igusa invariants. From the class numbers, we expect that the degree of the Igusa class polynomials should be of degree 1 (so there exists a hyperelliptic Jacobian defined over $\mathbb{Q}$ with CM by $\mathcal{O}_K$).

   Using the LLL-algorithm, we can get the minimal polynomials of the Igusa invariants, and in this case, they coincide with the modified Igusa class polynomial (here $N = 1$):

   $$[i_1 - 150660, i_2 N - 28343520, i_3 N - 9762768].$$

   The result suggests that the hyperelliptic Jacobian with CM by $\mathcal{O}_K$ should have the Igusa invariants $(150660, 28343520, 9762768)$. Magma suggests that such a hyperelliptic curve can be defined by the equation $y^2 = -x^5 - 3x^4 + 2x^3 + 6x^2 - 3x - 1$. This is isomorphic to the curve suggested in [vW99, Table 1].

3.7.2 Example 2: $\mathbb{Q}\left(\sqrt{-30+\sqrt{96}}\right)$

Todo: When finished, break down the bullet points...

1. We denote $K = \mathbb{Q}\left(\sqrt{-30+\sqrt{96}}\right)$. By testing small primes, we found from the splitting of $p = 5$ in K that, there exists hyperelliptic Jacobians defined over $\mathbb{F}_{p^2}$ with CM by $\mathcal{O}_K$.

2. We know that for a hyperelliptic curve $C/\mathbb{F}_{5^2}$, if we denote by n_C and n_J the number of points of C and $\mathrm{Jac}(C)$ over $\mathbb{F}_7$, then $\mathrm{Jac}(C)$ has CM by $\mathcal{O}_K$ only when $(n_C, n_J) = (16, 417), (24, 571), (28, 675)$ or $(36, 937)$.

3. We run through all possible Igusa invariants $(i_1, i_2, i_3) \in \mathbb{F}[\tau]/(\tau^2 + 4\tau + 2) \cong \mathbb{F}_5^2$, among a couple of possible choices, we can see that

$$C : y^2 = \tau^{13}x^6 + \tau^5 x^5 + \tau^{20}x^4 + \tau^{22}x + \tau^{19}$$

has the Jacobian with CM by $\mathcal{O}_K$.

4. Then, we need to compute the canonical lift of C. The first step is to compute its 10-theta null point. We need to start by computing the Rosenhein form of C and deduct the 2-theta null point from there. It turns out that the equation of C splits into linear factors in the field extension $\mathbb{F}_{5^{12}}$, which is the smallest field the Rosenhein invariants and the 2-theta null point lie in.

5. The next step is to compute the 10-theta null point $(a_{ij})_{0 \le i,j < 10}$, which involves solving 52 coordinates. Same as the previous example, we regroup the $2p^2 - 2$ unknown variables into $p + 1$ groups of $2(p - 1)$ variables. The F4 algorithm involved in step 2 of Algorithm 3.5.5 took about 10.5 seconds to compute a Gröbner basis of $2(p - 1)$ variables in the `grevlex` ordering. After that, to solve for the equations, as in step 3 of Algorithm 3.5.5, we applied the FGLM algorithm to convert the Grob̈er basis to the `lex` ordering. The FGLM algorithm took about 17.4 seconds to finish. For the final `lex` ordered Gröbner basis, the final generator is univariate in a_{45} of degree 306, with 12 roots of multiplicity 5^2. Those 12 roots will contribute to the coordinates of the 10-theta null point. We compute the 12 roots, which fall in the extension field $\mathbb{F}_{5^{24}}$, and found a match as in step 5 of Algorithm 3.5.5 almost immediately. Combining all the steps involving computing the 10-theta null point, it took us about 29 seconds in total.

6. After the 10-theta null point over $\mathbb{F}_{5^{24}}$ is computed, we need to pass to Algorithm 3.5.10 to compute the lifted theta null point over $Q_{5^{24}}$, the degree 24 unramified extension of $\mathbb{Q}_5$. As described in Algorithm 3.5.10, we will need to find 48 Riemann equations from Theorem 3.2.17 and 3 correspondence equations from Theorem 3.5.9 which forms a non-singular system of Artin-Schreier equation in the 51-dimensional projective space over $\mathbb{Q}_{5^{24}}$. After testing, we know that $N = 512$ 5-adic digits precision is sufficient to recover the Igusa polynomials after the LLL-reduction. The canonical lift step is the bottleneck and it took about 73 minutes to run through Algorithm 3.5.10.

7. From the lifted 2-theta null point, we can recover sequentially the Rosenhein invariants, the lifted hyperelliptic curve $\tilde{C}$, and the Igusa invariants $(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$. From the

class numbers, we expect that the degree of the Igusa class polynomials should be of degree 8. After applying the LLL-algorithm, we get in approximately 0.5 seconds that the irreducible factors of the modified Igusa polynomials of $(\tilde{i}_1, \tilde{i}_2, \tilde{i}_3)$ are $\{h_{K,1}, \widehat{h_{K,2}}, \widehat{h_{K,3}}\}$ where

$$\begin{aligned}
h_{K,1}(i_1) = &-5006007982921 i_1^8 + 71363156523747769882 08 i_1^7 - 103059741439702989326019669321 6 i_1^6 + 37019733797983009315008857459542891776 i_1^5 \\
&+ 1177131562420997519816364230135369385008102 4 i_1^4 + 379788558415335857863386892741701297861487715696 64 i_1^3 \\
&+ 56899977704119316148668396976362839865069280680558592 0 i_1^2 + 16710294186848614749610039434006818631272417866171825520 64 i_1 \\
&+ 14960121744868944695745562190420789362157636862456163970580 48, \\
\widehat{h_{K,2}}(i_1) = &(-15850179278777496034466422081189776 0 i_1^7 + 23327413057463058973733242059612989670157056 i_1^6 \\
&- 2040289838531044739300074750979686774291780646860 80 i_1^5 - 940693566406722395678412813864742417548601906390960889856 i_1^4 \\
&- 137611312008689296831809118392748720999736647711615696169462169 6 i_1^3 \\
&- 45159152499530136777973051621886886650928590741080316255000047976448 i_1^2 \\
&- 28670082556000074396017790052120317292453885028987974223745615116422348 8 i_1 \\
&- 42537301287478145892032628234737867648908057586577513299726268980041403596 8)/455091634811, \\
\widehat{h_{K,3}}(i_1) = &(-6621510601498418936962943217543072 72 i_1^7 + 113484641243574620712360609712684631732807232 i_1^6 \\
&- 2306289158690614751776608966639777541961309842657536 i_1^5 - 33560050799027478230163192004713741880742037509183478128 64 i_1^4 \\
&- 5035198581992372253485863299205548121579448500137586031066646528 i_1^3 \\
&- 162056862743043213054229143330119308828877290209215754586883268624384 i_1^2 \\
&- 103394367455175619953241355430607302543955992144934704017392661585815142 4 i_1 \\
&- 149080752025500493713409049501149205302615764953699876699701191428142740275 2)/5006007982921.
\end{aligned}$$

Check out Section 3.2.1.3 for the definition. For an i_1 satisfying $h_{K,1}(i_1) = 0$, for $n = 2, 3$, the corresponding i_n can be computed by $\widehat{h_{K_n}}(i_1)/h_{K,1}(i_1)$.

Since the degrees of the factors of the modified Igusa class polynomials fits the degree of the expected degree, we know that they are indeed the Igusa class polynomials.

3.8 The Complexity Analysis of the Main Algorithm

In this section, we will break down the main steps of the component algorithms in Sections 3.4, 3.5, and 3.6. By combining the complexity of each component, we will obtain the complexity statements in Theorem 3.1.1 and 3.1.2.

Before we start, we define two complexity constants that will be used throughout our analysis.

Definition 3.8.1. We define the following two constants, ω and μ, which measure the effectiveness of the implementations on basic objects.

(1) μ is the complexity constant so that multiplying two n-bit integers takes $O(n^\mu)$ bit operations, and so that multiplying two elements in $\mathbb{F}_{p^n}$ takes $O(n^\mu)$ arithmetics on $\mathbb{F}_p$ (so it is 2 for schoolbook multiplication, and $\log_2 3$ for the Karatsuba algorithm, etc).

(2) ω is the complexity constant for multiplying two $n \times n$ matrices (so it will be 3 for schoolbook multiplication, $\log_2 7$ for the Strassen algorithm, etc).

3.8.1 Issues on Curve Finding

The goal for this step is, given a primitive quartic CM field K, compute a finite field $\mathbb{F}_q = \mathbb{F}_{p^r}$ and a principally polarized abelian variety A over $\mathbb{F}_q$ satisfying $\mathrm{End}(A) \cong \mathcal{O}_K$. The algorithm in this step is described below.

We will use the following notations. For the principal quartic CM field K, let $K^\dagger$ be its reflex field, L be the algebraic closure, and let K_0 and $K_0^\dagger$ be the real quadratic subfield of K and $K^\dagger$, respectively. Also, let $D_0 = \Delta(K_0)$ be the discriminant of the real quadratic field, and let D_1 be the number satisfying $\Delta(K) = D_0^2 D_1$. The ultimate goal is to find the asymptotic complexity of finding the Igusa class polynomial via the p-adic method in terms of D_0 and D_1.

3.8.1.1 Finding the Underlying Finite Field

In this subsection, we will analyze step 2 in Algorithm 3.4.1. The complexity itself is negligible; instead, we need to estimate the size of the finite field $\mathbb{F}_q = \mathbb{F}_{p^r}$, as it is one of the most important parameters in the entire algorithm.

In practice, consider a principally polarized abelian surface $\tilde{A}$ over $\mathbb{Q}_q$, such that $\mathrm{End}(\tilde{A}) \cong \mathcal{O}_K$, we need its reduction $\tilde{A} \to A$ to $\mathbb{F}_q$ to be ordinary. Among all possible $q = p^r$, we need q to be as small as possible to speed up the searching of principal polarized abelian surfaces; and we also need p to be small, like the Gröbner basis approach to compute $2p$-theta null points seems to be unfeasible as p grows.

By the Cebotarev density theorem, case (a) occurs with probability $1/8$; and case (b) occurs with probability $1/4$. So roughly $3/8$ of the primes satisfies the splitting conditions, which is compatible with preliminary experiments. As all we need is the splitting condition described above, if we assume Heuristic 3.8.2 below, the average chosen p will be $O(1)$.

Heuristic 3.8.2. For a fixed prime p, the probability that p divides $\Delta(K) = D_0 D_1$ does not depend on the size of the discriminant of K.

Alternatively, assuming GRH, we can also potentially give some upper bound arguments. For instance, [LO77] gave a upper bound on the smallest prime which splits completely in a number field: if p is the smallest such prime, there exists a efficiently computable constant c satisfying $p \leq c(\log \Delta(K))^2 = c(\log(\tilde{O}(D_0^2 D_1)))^2$. For newer results, see [GMP19]. Our application allows some cases that the prime does not need to split completely, so the upper bound could be decreased.

After the characteristic of the finite field, p is decided, we need to decide the degree of extension $q = p^r$. Then by [Koh], we can use Lemma 3.4.3 to choose r.

To estimate the minimal r, the heuristic is that given an r dividing $h(K)$, the class number of K, the probability that $\mathbb{F}_{p^r}$ is the smallest possible extension equals the probability that a random element in $\mathrm{Cl}(K)$ has order r. Preliminary experiment results show that it should be true for at least most of the non-abelian cases, but there are counterexamples for cyclic cases:

Example 3.8.3. Let $K \cong \mathbb{Q}[x]/(x^4 + 41x^2 + 164)$ be a cyclic extension of $\mathbb{Q}$. The real subfield of K has discriminant $D_0 = 41$. It is readily computed that $\mathrm{Cl}(K) \cong \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$, so a random element in $\mathrm{Cl}(K)$ has a $1/4$ chance to have order 1 and a $3/4$ chance to have order 2. However, experimental results showed that the probabilities that $r = 1$ and $r = 2$ are both $1/2$.

More work needs to be done to characterize the situations like Example 2. However, under the following heuristic assumption, we can assert that for the average case, $r = O(h(K))$.

Heuristic 3.8.4. For a fixed quartic CM field K. When varying through all admissible p (such that ordinary abelian surfaces over $\overline{\mathbb{F}_p}$ with CM by $\mathcal{O}_K$ exists) and varying through all prime ideals $\mathfrak{p}$ over p, the ideal class $[\mathfrak{p}] \in \mathrm{Cl}(K)$ distributes uniformly in a subgroup G of $\mathrm{Cl}(K)$ with a small index.

Another issue is that we want to deal with the class number $h(K)$. We may simply treat it as a parameter since it also plays an important role as the degree of the Igusa class polynomial, Alternatively, we can also try to bound it in terms of D_0 and D_1.

Denote $h^-(K) := |\mathrm{Cl}(K)|/|\mathrm{Cl}(K_0)|$ the relative class number. Louboutin gave a bound in [Lou03], saying that $h^-(K) = \tilde{O}(\sqrt{D_0 D_1})$. For the class number of the real quadratic field K_0, Le provided a bound in [Le94], saying that $h(K) = O(\sqrt{D_0})$. Combining these, we will have $h(K) = \tilde{O}(\sqrt{D_0^2 D_1})$. However, to obtain a better average bounds on r, we want to have the structure of $\mathrm{Cl}(K)$ as well. We collect our results in the following lemma.

Lemma 3.8.5. In Algorithm 3.4.1, under Heuristics 3.8.2 and 3.8.4 and GRH:

1. In average, p is in $O(1)$, and r is in $O(h(K))$. If we estimate $h(K)$ in terms of D_0 and D_1, then $r = \tilde{O}(\sqrt{D_0^2 D_1})$.

2. For the worst case, $p = \log(\tilde{O}(D_0^2 D_1))$, and $r = O(h(K)) = \tilde{O}(\sqrt{D_0^2 D_1})$.

However, since p and r are the most important parameters in the algorithm, in the rest of the chapter, we will avoid using rough estimates as in Lemma 3.8.5.

3.8.1.2 Finding a Curve over Given Finite Field via Computing Endomorphism Rings

For the remaining subsection, we will put aside the estimates of $q = p^r$ and find complexity estimates using q as a parameter. Finding a suitable hyperelliptic Jacobian consists of looping over steps 5, 6, and 7 in Algorithm 3.4.1. Our complexity estimation follows the ideas in [BGL11], except that we are working over the extension field $\mathbb{F}_q$, while [BGL11] put stronger restrictions on the splitting of p so that they can work over the prime field $\mathbb{F}_p$.

We first analyze the complexities of step 5. The first goal is to estimate the number of iterations we need to make in the for loop. Before step 5, there will be $q^3 = p^{3r}$ possible curves, since we can run through all isomorphism classes of principally polarized hyperelliptic Jacobians of genus 2 by running through the Igusa or G2 tuples defined over $\mathbb{F}_q$. After step 6, there will be $|\mathfrak{C}(K)|$ Jacobians remaining, where $\mathfrak{C}(K)$ is the Shimura class group of K, and the size can be computed via Corollary 3.2.9. From the corollary, the size of $|\mathfrak{C}(K)|$ is a constant times $h^-(K) = \tilde{O}(\sqrt{D_0 D_1})$. Intuitively, on average one needs to test $O(q^3/\sqrt{D_0 D_1})$ hyperelliptic curves before one obtains one with the correct endomorphism ring.

For the complexity of looping over step 5, one needs to apply Mestre's algorithm. For the explicit formulas involved, see [LY11, Appendix A.2]. Note that the complexity here should be absorbed by looping over step 6, where we will describe next.

For step 6, we need to compute the endomorphism algebra for a given Jacobian curve, and this is essentially the same as computing $|\mathrm{Jac}(\mathcal{C})[\mathbb{F}_q]|$ and $|\mathrm{Jac}(\mathcal{C})[\mathbb{F}_{q^2}]|$, or equivalently the characteristic polynomial of the Frobenius. We can use the theory as described in [JW15]. Suppose $A = \mathrm{Jac}(\mathcal{C})$ be a principally polarized abelian surface over $\mathbb{F}_q$ of dimension 2. By the method of Pila, it takes $O((\log q)^9)$ time to compute the characteristic polynomial of the Frobenius. Multiplying by $q^3/\sqrt{D_0 D_1}$, the average number of the curve we need to consider in step 1, we obtain the following Lemma:

Lemma 3.8.6. Taking the number of loops into account, the time complexity for steps 5 and 6 in Algorithm 3.4.1 are

$$O(q^3(\log q)^9/\sqrt{D_0 D_1}).$$

Now we estimate the running time for step 7. One key factor is the number of candidate curves that passed through step 6, so we need to look at 6 in more detail. The Frobenius polynomial we computed in step 6 will be of the form $x^4 + s_1x^3 + (s_2 + 2q)x^2 - qs_1x + q^2$. s_1 and s_2 are related to the number of F_q-points on the curve $\mathcal{C}$ and on the Jacobian $A = \mathrm{Jac}(\mathcal{C})$ via the following formula:

$$s_1 = |\mathcal{C}(\mathbb{F}_q)| - q - 1;$$
$$s_2 = \frac{s_1^2 + 2|A(\mathbb{F}_q)| - |\mathcal{C}(\mathbb{F}_q)|^2 - q^2 - 2q - 1}{2}.$$

There are several known bounds on the values s_1 and s_2:

Proposition 3.8.7 ([JW15]). Let q, s_1 and s_2 be defined as above.

1. (The Weil bound) $|s_1| \leq 4\sqrt{q}$, and $|s_2| \leq 4q$.
2. (The Ruck bound) $s_1^2 - 4s_2 > 0$, and $s_2 + 4q > 2|s_1|\sqrt{q}$.

In addition, we make the following assumption (more experiments need to justify the validity):

Heuristic 3.8.8. (s_1, s_2) distributes uniformly within the region described by the Weil and Ruck bounds.

Since there are only 1 or two (s_1, s_2) pairs which leads to the correct endomorphism algebra, the chance that a random abelian surface passes through step 1 is $O(q^{-3/2})$, and on average, the number of curves entering step 2 will be $O(q^{3/2}/\sqrt{D_0D_1})$.

Remark 3.8.9. A more careful approach to compute the number of curves passing through step 1 given in [BGL11]. The key idea is to note that if $\mathrm{End}(A) \otimes_{\mathbb{Q}} \mathbb{Z} \cong K$ has the correct endomorphism algebra, then we know that $\mathbb{Z}[\pi, \overline{\pi}] \subseteq \mathrm{End}(A) \subseteq \mathcal{O}_K$. The alternate method is then to find all possible orders for $\mathrm{End}(A)$ bounding between $\mathbb{Z}[\pi, \overline{\pi}]$ and $\mathcal{O}_K$, and then estimate the class number for all the intermediate orders.

For the complexity in looping over step 7, the bottleneck is the complexity of computing endomorphism rings. The more classical method by Freeman and Lauter in [FL08] has complexity $O((\log q)^{18})$; and the more recent method proposed [Bis15, Spr19] has complexity

$$L\left[\frac{1}{2}\right](q)^{2\sqrt{3}+o(1)} = \exp\left(\sqrt{\log q \cdot \log\log q}\right)^{2\sqrt{3}+o(1)}.$$

The following Lemma integrates Lemma 3.8.6, Heuristic 3.8.8, and the complexity of endomorphism rings computation as described above:

Lemma 3.8.10. Assuming Heuristic 3.8.8, and denote by X the complexity of endomorphism ring computation algorithm (So $X = (\log q)^{18}$ in Freeman-Lauter [FL08], and $X = L[1/2](q)^{2\sqrt{3}}$ in Bisson [Bis15]). The complexity of step 2 as a whole (counting loops) is

$$O\left(\frac{q^3(\log q)^9}{\sqrt{D_0 D_1}}\right) + O\left(\frac{q^{3/2}X}{\sqrt{D_0 D_1}}\right).$$

The bottleneck is still the q^3 term in step 6. If we cannot cut that down the lifting method would turn out to be infeasible as long as q reaches a certain value.

3.8.2 From 2-theta Null Points to $2p$-theta Null Points

In this section, we will estimate the complexity of steps 1 to 5 in Algorithm 3.5.5. Note that not only the time complexity, the space complexity is also important, which is indeed the bottleneck in our implementation.

We first analyze step 1, which involves the reduction of Riemann relations. Denote $Z(\overline{n})$ as the group $(\mathbb{Z}/n\mathbb{Z})^2$, and for $u = (i,j) \in Z(\overline{2p})$, let $b_u = b_{ij}$. And for all $n = dm$, we consider $Z(\overline{m})$ as a subgroup of $Z(\overline{n})$ by the embedding $(i,j) \mapsto (di, dj)$.

Under this notation, recall that the Riemann relations are of the following form:

$$\sum_{t \in Z(\overline{2})} \chi(t) b_{x+y+t} b_{x-y+t} \sum_{t \in Z(\overline{2})} \chi(t) b_{u+v+t} b_{u-v+t} = \\ \sum_{t \in Z(\overline{2})} \chi(t) b_{x+u+t} b_{x-u+t} \sum_{t \in Z(\overline{2})} \chi(t) b_{y+v+t} b_{y-v+t},$$

where $\chi \in Z(\overline{2})^D$ is a character of $Z(\overline{2})$, and $x, y, u, v \in Z(\overline{2p})$ are congruent to $Z(\overline{p})$. Since we only consider Riemann relations in $\mathbb{F}_q[\{b_{ij}\}_{0 \le i \le p, j=0,p}]$ at this stage, we really only need to take x, y, u, v be of the form $(i, 0) \in Z(\overline{2p})$. There are $2p^4$ choices for x, y, u, v and 4 choices for χ, yielding $8p^4$ possible Riemann relations.

Among the aforementioned $2p^4$ Riemann relations which are all degree 4 homogeneous polynomials in $\mathbb{F}_q[\{b_{ij}\}_{0 \le i \le p, j=0,p}]$, we need to know how many of them are $\mathbb{F}_p$-linearly independent. It is only known by experiments that there are 5, 30, and 114 linearly independent Riemann relations when $p = 3, 5$ and 7, respectively, and in practice, it is not yet computable for larger p due to memory constraint. However, we have the following asymptotic estimation:

Lemma 3.8.11. For an odd prime p, there are $O(p^4)$ $\mathbb{F}_p$-linearly independent Riemann relations in $\mathbb{F}_q[\{b_{ij}\}_{0\le i\le p, j=0,p}]$.

Proof. $F_q[\{b_{ij}\}_{0\le i\le p, j=0,p}]$ contains $2p+2$ variables, and Riemann relations are homogeneous polynomials of degree 4. Let M be the set of degree 4 monomials, then $|M| = \binom{2p+2+4-1}{4}$. Denote by $\mathcal{V}$ the $\mathbb{F}_p$-vector space spanned by the Riemann relations. By observation, each degree 4 monomial occurs in some Riemann relation, hence for any proper subset $M' \subsetneq M$, $\mathcal{V} \cap \bigoplus_{m\in M'} \mathbb{F}_p m \neq \{0\}$. Let B be a basis of $\mathcal{V}$ which is a collection of Riemann relations. Let $B(M)$ be the monomials involved in the elements of B. Then we know that $B(M) = M$. But it is clear from the form of Riemann relations that element in B can contain at most 32 monomials, so $|M| = |B(M)| \le 32|B|$. Hence $\dim \mathcal{V} = |B| \ge \binom{2p+5}{4}/32 = O(p^4)$. □

Now we give statements on the time and space complexity for step 1. The essence for step 1 is to find a minimal spanning set of $\mathcal{V} \in \mathbb{F}_p^N$, where $N = \binom{2p+5}{4} = O(p^4)$, the generators of $\mathcal{V}$ is of size $8p^4 = O(p^4)$, and $\dim \mathcal{V} = O(p^4)$. While computing the minimal spanning set of $\mathcal{V}$, if Gaussian elimination is used (which seems to be the most inefficient), the time complexity is $O(p^{12})$ and the space complexity is $O(p^8)$ entries of size $O(1)$. If the sparse matrix structure is used, then it is possible to reduce to $O(p^8)$ in time complexity and $O(p^4)$ space complexity (verification needed). We conclude our current result in the following lemma:

Lemma 3.8.12. For Step 1 in Algorithm 3.5.5, the time complexity is at most $O(p^{12})$, and the space complexity is at most $O(p^8)$.

Now we analyze Step 2. The core content involves computing a Gröbner basis under the `grevlex` monomial ordering with priority. It is still hard to estimate the improvements made by applying the priority on the variables U in Step 2, so, at this stage, we just try to provide general arguments on Faugére's F4/F5 algorithm. We will apply the following Proposition, which turns out to be close enough to our scenario:

Proposition 3.8.13 ([BFS15, Proposition 1]). Let $(f_1, \cdots, f_m)$ be a system of homogeneous polynomials in $k[x_1, \cdots, x_n]$. Then the number of operations in k required to compute a Gröbner basis for a `grevlex` order is $O\left(mD\binom{n+D-1}{n}^{\omega}\right)$ as $D \to \infty$, where D is the degree of the ideal generated by $(f_1, \cdots, f_m)$, and ω is as in 3.8.1.

Note that besides the issue of prioritizing, our scenario still differs from Proposition 3.8.13 because after evaluating $(b_{00}, b_{0p}, b_{p0}, b_{pp})$ by $(a_{00}, a_{01}, a_{10}, a_{11})$, the ideal generated by the Riemann relations is no longer homogeneous. However we can still use Proposition 3.8.13 to obtin an upper bound of the time complexity by interchanging the evaluation and the F4 (or F5) algorithm.

For the parameters in Proposition 3.8.13, it is immediate from Algorithm 3.5.5 and Lemma 3.8.12 that we have $n = 2p + 2$ variables and $m = O(p^4)$ equations. It remains to determine D, the degree of the ideal. Experimental results showed that, generically after fixing $(b_{00}, b_{0p}, b_{p0}, b_{pp}) = (a_{00}, a_{01}, a_{10}, a_{11})$, all the other variables will be uniquely determined by b_{10} (or any other variable in $U \cup V$ by symmetry). And the unique univariate polynomial in the Gröbner basis in Step 3 for b_{01} has 1 root of multiplicity 1, 1 root of multiplicity p and $(p^2 - 1)/2$ roots of multiplicity p^2. Therefore, $D = p^2 \cdot (p^2 - 1)/2 + p + 1 = (p^3 - p^2 + 2)(p + 1)/2 = O(p^4)$.

Plugging all variables back into Proposition 3.8.13, we see that the time complexity for the `grevlex` Gröbner basis algorithm is

$$O\left(mD\binom{n + D - 1}{n}^{\omega}\right) = O(p^4 p^4 (p^4)^{(2p+2)\omega}) = O(p^{8(\omega(p+1)+1)}).$$

Lemma 3.8.14. For Step 2 in Algorithm 3.5.5, the time complexity is at most $O(p^{8(\omega(p+1)+1)})$ operations in $\mathbb{F}_q$.

We remark that it seems that many modifications are done beyond the scenario in Proposition 3.8.13, so the bound above could be largely optimized.

Next, we analyze step 3, which computes the Gröbner basis in the lexicographical $<_{\texttt{lex}}$ order. Since we used Magma's implementation on the FGLM algorithm as described on [FGLM93], we will use the complexity arguments there, and then describe some possible improvements.

As in Algorithm 3.5.5, we need to convert a `grevlex` Gröbner basis $(G_V, <_{\texttt{grevlex}})$ in 4 variables to a lexicographical Gröbner basis $(G_2, <_{\texttt{lex}})$. When the ideal generated is of dimension 0, this can be handled by the FGLM algorithm, and we briefly summarize the algorithm as follows. Let $M = \mathbb{F}_q[x_1, \cdots, x_n]/\langle G_V \rangle$ be the dimension D $\mathbb{F}_q$-vector spaces determined by the first Gröbner basis. Also, let $B = \{\epsilon_1, \cdots, \epsilon_D\}$ be the a basis of M, where ϵ_i are monomials ordered increasingly by $<_{\texttt{grevlex}}$. For each $\epsilon_j \in B$, sequentially in `lex` compute the normal form of $\epsilon_j x_i$ with respect to the Gröbner basis G_V for $1 \leq i \leq n$. Then we fit the coefficients of the normal form into column vectors of a matrix. When there are linear dependencies among the column vectors, we get an element in the new Gröbner basis $(G_2, <_{\texttt{lex}})$. We see that linear algebra plays the central role in the FGLM algorithm, which allows us to compute the time and space complexity. In particular, we have the following Proposition:

Proposition 3.8.15 ([FGLM93, Theorem 5.1])**.** When the ground field of the polynomial ring is a finite field, the time complexity is $O(nD^3)$ and the space complexity is $O(D^2)$, where n is the number of variables and D is the degree of dimension 0 ideal.

Plugging in $n = 4$ and $D = O(p^4)$ (from the analysis of Step 2), we have the following result:

Lemma 3.8.16. For Step 3 in Algorithm 3.5.5, the time complexity (number of $\mathbb{F}_q$ operations) is at most $O(p^{12})$, and the space complexity is at most $O(p^8)$.

Remark 3.8.17. It seems that this is the best the built-in FGLM implementation Magma can do so far. Indeed Faugére and Mou proposed in [FM17] the sparse version of the FGLM algorithm, which has the potential to outperform the original version in Magma. Need to understand more about this.

Next we analyze step 4, which involves solving the $(2n-2)$ variables of the $2p$-theta null points from the Gröbner bases $(G_U, <_{\texttt{grevlex}})$ and $(G_2, <_{\texttt{lex}})$. The most costly step here is to solve for the first variable b_{01} from the first polynomial in the Gröbner basis $f(b_1)$ over its splitting field (the size of the splitting field is unknown). The polynomial f has degree D, and contains 1, 1, and $(p^2-1)/2$ roots of multiplicity 1, p and p^2, respectively. Hence it is of negligible time to derive from f to the polynomial $\tilde{f}$, the square-free polynomial whose roots are exactly those of f with multiplicity p^2 (hence $\deg \tilde{f} = (p^2-1)/2$).

Classically (see [Coh13, Section 3.4]), solving $\tilde{f}$ over its splitting field contains two stages. The first stage is to factor $\tilde{f}$ into polynomials of the form f_r, where f_r is the product of all degree r polynomials which divides $\tilde{f}$ and is irreducible over $\mathbb{F}_q$. The second stage is to solve f_r for each r, using the Cantor-Zassenhaus algorithm for instance.

Suppose the splitting field is a degree d extension of $\mathbb{F}_q$. In stage 1, for each $r \mid d$, we indeed have $f_r = \gcd(\tilde{f}(b_{01}), b_{01}^{q^r} - b_{01})$. The complexity for this is $O(r(\deg \tilde{f})^2(\log q + \deg \tilde{f})) = O(rp^4(\log q + p^2))$.

The Cantor-Zassenhaus stage has complexity $O((\deg \tilde{f})^2 \log q \log r) = O(p^4 \log q \log r)$. Combining all possible r, which ranges through divisors of d, we get the overall complexity $O(\sigma(d)p^4(\log q + p^2))$.

At this moment, not much is known about the splitting field extension degree d, except that when $p = 3$, Carls gave in [Car10] that $d \mid 48$. It is backed up by experiments that in most cases d is small, so both theoretically and by experiment, we can assume that Step 4 takes negligible time compared to other steps in Algorithm 3.5.5.

Finally, we analyze step 5. From step 4, we have a solution set S, which contains $(p^2-1)/2$ sets of quadruples. From step 4, we also partitioned S into $p+1$ groups, and each group contains $(p-1)/2$ quadruples. On the other hand, for the $2p$-theta null point $(b_{ij})_{0\leq i,j<2p}$, we can separate the unknown positions into $(p^2-1)/2$ quadruples, and the quadruples form $p+1$ groups, with each group consists of $(p-1)/2$ quadruples.

We need to fill in the positions in (b_{ij}) by the quadruples in S, by the following rules: Each group in S has to be filled into a group in (b_{ij}); and within a group of (b_{ij}), when a quadruple in (b_{ij}) is fixed by a quadruple in S, the other $(p-3)/2$ quadruples in (b_{ij}) are automatically fixed. Therefore, there are $(p+1)!$ assignments for the groups and $(p-1)/2$ assignments for the quadruples in each group, yielding $(p+1)!((p-1)/2)^{p+1}$ assignments in total.

Among all these assignments, only $|\mathrm{GL}_2(\mathbb{F}_p)|/|\{\pm 1\}| = p(p-1)^2(p+1)/2 = O(p^4)$ of them are the desired $2p$-theta null points. Since we randomly choose the assignments and test the validity, and testing takes $O(1)$ arithmetic in the ground field, In average it takes $O((p-2)!((p-1)/2)^p)$ field operations to get a valid $2p$-theta null point. We summarize the result in the following lemma.

Lemma 3.8.18. In Step 5 of Algorithm 3.5.5, suppose that the $2p$-theta null point lies on $\mathbb{F}_{q^d}$, on average it takes

$$O\left((p-2)!\left(\frac{p-1}{2}\right)^p\right)$$

field operations on $\mathbb{F}_{q^d}$, or $O\left(d^\mu (p-2)!\left(\frac{p-1}{2}\right)^p\right)$ field operations of $\mathbb{F}_q$ to derive a valid $2p$-theta null point.

To summarize Algorithm 3.5.5, we see that for a general p, the bottlenecks in terms of time complexity are Steps 2 and 5, which are exponential in p; the bottlenecks in terms of space complexity are Steps 1 and 3, which are $O(p^8)$ (although there is a potential to improve by applying sparse matrices).

3.8.3 From $2p$-theta Null Points over Finite Fields to $2p$-theta Null Points over Local Fields

In this section, we will look at the complexity of Algorithm 3.5.10. Carls and Lubicz provided a complexity formula for this in [CL09, Section 4.2], which follows from the ones in Lercier and Lubicz in [LL06, Algorithm 5.1]. Since the scenario in [LL06] is for fixed p (there $p = 2$), so we need look closer to the formula.

It is not hard to see that Step 3 is dominated by the highest precision iteration, and in that iteration, the dominating step is to solve the Artin-Schreier equation. Adapting the analysis from [LL06, Algorithm 5.1] and substitute all operations on $\mathbb{F}_2$ (respectively $\mathbb{Q}_{2^d}$) to operations on $\mathbb{F}_p$ (respectively $\mathbb{Q}_{p^d}$), we get the following lemma:

Lemma 3.8.19. The complexity for Algorithm 3.5.10 is $O(\log(N)N^{\mu}d^{\mu})$ operations on $\mathbb{F}_p$, where:

- d is the extension where the $2p$-theta null point lies in. When we solve the Gröbner basis system to solve the $2p$-theta null point, we will potentially need to get to a larger finite field extension. Carls in [Car10] claimed that when $p = 3$, the maximal extension needed will be $d = 48r$, where r is the extension where the 2-theta null point (though in most case d will be less or equal to $3r$).
- The complexity constants μ is as in Definition 3.8.1
- N is the required precision.

We note that N is affected by many factors, such as the coefficient bounds and denominator bounds of the Igusa class polynomial, and the error factor of the LLL algorithm. We will figure out the asymptotic size of N in section 3.8.4.2.

3.8.4 Recovering Igusa Class Polynomials

For this step, the input will be the Igusa invariants (i_1, i_2, i_3) in the local field $\mathbb{Q}_p$, and the output will be a factor of the Igusa polynomial, which is the minimal polynomial of the factors. If it were the elliptic curve case, there will be two possible approaches.

3.8.4.1 Using the Actions of Ideal Classes $[\mathfrak{a}] \in \mathrm{Cl}(K)$

This is the best method in genus 1 and suggested as an improvement for the CRT method in the genus 2 case in the paper [BGL11]. However, it does not seem that the method can be applied to our scenario. First, we need the actions on abelian surfaces over $\mathbb{Q}_q$ rather than a finite field; and in addition, the modular polynomial of the Siegel modular variety $V(f;l)$ described in section 2.3 is hard to compute for larger l.

3.8.4.2 Using the LLL algorithm to Find Minimal Polynomials

We will discuss the complexity of Algorithm 3.6.1. Let $\mathfrak{C}(K)$ be the Shimura class group, $d := [\mathbb{Q}_q : \mathbb{Q}_p]$ be the extension degree, and N be the required precision. Also, we take s to be the degree of the minimal polynomial of the Igusa invariants. In general, we will have $|\mathfrak{C}(K)| = s$ or $2s$.

We first explain the validity of Algorithm 3.6.1. First, we show that there exists a vector of the form $\mathbf{v} := [m_0\ m_1\ \cdots\ m_s - \epsilon_0\ \cdots\ -\epsilon_{d-1}]$ in the left kernel Λ of M. Suppose that $f(x) = \sum_{i=0}^{s} m_i x^i$ be the minimal polynomial, we have $\sum_{i=0}^{s} m_i a_{i,j} w_j \equiv 0 \pmod{p^N}$. Denote $\sum_{i=0}^{s} m_i a_{i,j} w_j = \epsilon_j p^N$ for each $0 \leq j < d$, then $[m_0\ m_1\ \cdots\ m_s - \epsilon_0\ \cdots\ -\epsilon_{d-1}]M = 0$. And this justify that $[m_0\ m_1\ \cdots\ m_s - \epsilon_0\ \cdots\ -\epsilon_{d-1}] \in \Lambda$.

Next, we show that when the precision N is sufficiently large, $[m_0\ m_1\ \cdots\ m_s - \epsilon_0\ \cdots\ -\epsilon_{d-1}]$ is indeed the shortest vector and can be recovered by the LLL algorithm, by proving the following lemma:

Lemma 3.8.20. Let C be the upper bound for both the denominator and the coefficients of the Igusa class polynomial. Suppose N is chosen so that $N = \log_p(O(2^{(s+1)/2} C^2 s^2 d^2)))$, then the shortest vector provided after Step 2 of Algorithm 3.6.1 gives the correct coefficients of the minimal polynomial.

Proof. We first estimate the norm of the vector $\mathbf{v} = [m_0\ m_1\ \cdots\ m_s - \epsilon_0\ \cdots\ -\epsilon_{d-1}]$. The first $s+1$ entries are from the minimal polynomial, hence bounded by C. For the last d entries involving ϵ, we have $|\epsilon_j| \leq (1/p^N) \sum_{i=0}^{s} |a_{i,j} m_i| \leq \sum_{i=0}^{s} m_i \leq (s+1)C$. Therefore, $||\mathbf{v}|| = O(Csd)$.

We then estimate the lengths of other vectors in the left kernel Λ of M which are not scalar multiples to $\mathbf{v}$. Let $\mathbf{u} := [u_0 \cdots u_{d+s}] \in \mathbb{Z}^{d+s+1}$ be such a vector. Then $\sum_{i=0}^{s} u_i \iota^i \equiv 0 \pmod{p^N}$ for the Igusa invariant ι, which means that there exists embeddings $\tilde{u}_0, \cdots, \tilde{u}_s \in \mathbb{Z}_p$ of $u_0, \cdots, u_s$ satisfying $\sum_{i=0}^{s} \tilde{u}_i \iota^i = 0$. Hence there exists a $\tilde{c} \in \mathbb{Z}_p$ such that $[\tilde{u}_0 \cdots \tilde{u}_s] = \tilde{c}[m_0 \cdots m_s]$. This means that projecting $\tilde{c}$ into the $\mathbb{Z}/p^N\mathbb{Z}$ and take the least non-negative representative in $\mathbb{Z}$, we have $\mathbf{u} = c\mathbf{v} + p^N \mathbf{e}$ for some $\mathbf{e} \in \mathbb{Z}^{d+s+1}$.

We claim that the second shortest vector $\mathbf{w}$ should have $||\mathbf{w}|| = O(p^N/Csd)$. For $i = 0, \cdots, s$, let $\mathbf{b}_i = [0, \cdots 0, p^N, 0, \cdots, 0, -a_{i,0}, \cdots, -a_{i,d-1}]$ (p^N is in the $(i+1)$-th entry). Observe that the vectors $\mathbf{b}_0, \cdots, \mathbf{b}_s$ together with $\mathbf{v}$ span the left kernel Λ. If we only keep the first $(s+1)$-entries, then $\mathbf{b}_0, \cdots, \mathbf{b}_s$ has volume $(p^N)^{s+1} = p^{N(s+1)}$. Joining $\mathbf{v}$, then $p^N\mathbf{v}$ is the smallest scalar product of $\mathbf{v}$ which is in $\operatorname{span}(\mathbf{b}_0, \cdots, \mathbf{b}_s)$. Therefore, we know that restricted in the first $(s+1)$-entries, Λ has volume p^{Ns}. Since there are $s-1$ linearly independent vectors of norm p^N, the norm of the second shortest vector $\mathbf{w}$ must have size at least $O(p^N/Csd)$ putting the last d-entries back will only lengthen $\mathbf{w}$.

There is an adjustable parameter $\delta \in (0.25, 1)$ in the LLL algorithm, which controls the Lovász condition and the "idealness" of the output short lattice base. Suppose we choose, as the default of Magma, $\delta = 0.75$, then the classical result states that if $2^{(s+1)/2}||\mathbf{v}|| < ||\mathbf{w}||$, the first vector in the output of the LLL algorithm will be indeed $\mathbf{v}$.

Therefore, given $N = \log_p(O(2^{(s+1)/2}C^2s^2d^2)))$, we will obtain the shortest vector $\mathbf{v}$ containing the coefficients of the minimal polynomial. □

The next issue is to estimate $C := \max\{|m_0|, \cdots, |m_s|\}$, the maximal coefficient of the minimal polynomial and the denominator. And suppose we have the bound on C, then we will get a bound on N as well.

Lemma 3.8.21 ([BGL11, Section 6.4])**.** $C = \exp(\tilde{O}(D_0^{5/2}D_1^{3/2}))$, and $N = \tilde{O}(D_0^{5/2}D_1^{3/2})$.

Proof. For the denominator (defined as the least common multiple of the denominators of the coefficients of the Igusa class polynomial when we write it as a monic polynomial) in the Igusa class polynomial, Streng have in [Str14] the following theorem, based on the result of Goren and Lauter:

Theorem 3.8.22 (Streng, [Str14, Theorem 10.1])**.** Let $K = \mathbb{Q}(\sqrt{-a + b\sqrt{d}})$ with $d = D_0$, and $a < 8\sqrt{D_0D_1}/\pi$. Then the denominator of the Igusa class polynomial divides $D = 2^{24h'}\mathcal{D}_1^2$, where $h' = |\mathfrak{C}(K)|$,

$$\mathcal{D}_1 = \left(\prod_{\substack{p<4da^2 \\ p \text{ prime}}} p^{\left\lfloor 4f(p)\left(1+\frac{1\log(2da^2)}{\log p}\right)\right\rfloor} \right)^{h'}.$$

Here $f(p)$ is 3 if $p \leq 3$ and ramifies in K, and 1 otherwise.

From Theorem 3.8.22, it is deducted in [Str10, section II.9] that the denominator is $\exp(\tilde{O}(D_0^{5/2}D_1^{3/2}))$. Moreover, in [Str10, section II.11], Streng also proved that the maximal absolute value of the monic Igusa class polynomial is also bounded by $\exp(\tilde{O}(D_0^{5/2}D_1^{3/2}))$. Combining both bounds we have $C := \max\{|m_0|, \cdots, |m_s|\}$.

Compare with the estimate of s, which is related to the relative class number $h^-(K)$, we have $s = \tilde{O}(D_0^{1/2}D_1^{1/2})$. So both occurences of s will also be absorbed. Also, since d is also polynomial in D_0 and D_1, it is also absorbed in the estimation of N. This means that $N = \tilde{O}(D_0^{5/2}D_1^{3/2})$ is dominated by the contribution of C. □

Remark 3.8.23. In [LV15], Lauter and Viray gave a sharper bound for the denominator of the Igusa class polynomials, which used results on certain intersection formulas on Hilbert modular surfaces. It could potentially improve the bound in Lemma 3.8.21, but as the formula was extremely lengthy, so we decided not to address it in the lemma.

Finally, for the complexity of the LLL algorithm, [GHK+06], the L^2 variant of the general LLL algorithm has the complexity $O((s+d)^5(s+d+N)N)$ and can be reduced to

$$O((s+d)^4(s+d+N)N)$$

in our setting. Combining our estimates for d and N, we have the following:

Lemma 3.8.24. The complexity for using the LLL algorithm to recover a factor of the Igusa class polynomial is $\tilde{O}(D_0^7D_1^5)$ (in terms of $\mathbb{Z}$-operations).

As we have complexity statements from all the steps now, by assembling all the parts, we can see that the main Theorems 3.1.1 and 3.1.2 follows from Lemmas 3.8.10, 3.8.14, 3.8.18, 3.8.19, and 3.8.24.

3.9 Future directions

To reduce the complexity of Gröbner basis computation so that the computation of canonical lifting works out for as many prime p as possible, we made observations in Section 3.5.3. However, we are not yet able to rigorously justify our observations in this dissertation. Some facts which are believed to be helpful regarding the actions of $\mathrm{Aut}_{\mathbb{G}_m} H_{\overline{2p}}$ on valid theta $2p$-mull points are given in [FLR11]. There is still a gap to apply the actions [FLR11] directly, since their results needed the theta level to be not divisible by p, which is not our case. We are still working out to get a better understanding of the tools required for Section 3.5.3.

Nevertheless, as p increases, the size of the Gröbner bases expand exponentially, and the extension field degree of the $2p$-theta null points also increased with a speed that is hard to control. Is there any correspondence relation for the p-adic canonical lifting which does not rely on the $2p$-theta null points? A positive answer to the question might lead to a breakthrough for the p-adic method. At this moment, for $p = 3$, [CKL08] found a relation on the 4-theta null point, but we do not know anything beyond that.

Appendix I
A Proposal of a Signature Scheme in Genus 2

In this appendix, we provide an outline of a signature scheme, which generalized the key component of the signature scheme by Galbraith, Petit and Silva in [GPS19, Section 4] to superspecial abelian surfaces. The goal of this appendix is to provide a connection to the principal ideal problem with a signature scheme over superspecial ableian surfaces.

1 A sketch of Galbraith et al.'s signature scheme for supersingular elliptic curves

Galbraith's scheme relied on the fact that the following problems are computationally hard. Indeed, it is shown in [EHL$^+$18] that these hard problems are equivalent.

- `MaximalOrder`: Given a supersingular curve E defined over $\mathbb{F}_{p^2}$ and a basis of the quaternion algebra $B_{p,\infty}$, find a basis $\{\beta_1, \beta_2, \beta_3, \beta_4\} \subseteq B_{p,\infty}$, so that $\mathrm{End}_{\overline{\mathbb{F}_p}}(E) \cong \mathbb{Z}\beta_1 + \mathbb{Z}\beta_2 + \mathbb{Z}\beta_3 + \mathbb{Z}\beta_4$.

- `EndomorphismRing`: Given a supersingular j-invariant $j \in \mathbb{F}_{p^2}$, output $\mathrm{End}_{\overline{\mathbb{F}_p}}(E(j))$, which is represented by the four rational maps $E \to E$ generating the endomorphism ring.

- ℓ-`PowerIsogeny`: Given a prime p and two supersingular elliptic curves E_1, E_2 defined over $\overline{\mathbb{F}_p}$, and a small prime $\ell \neq p$, output an ℓ-power isogeny $\phi : E_1 \to E_2$, where $\deg \phi = \ell^k$, represented by a chain of ℓ-isogenies of length k.

On the other hand, Galbraith's scheme required efficient algorithms for the following questions. We list their problems alongside with a reference to the algorithms.

- `IsogenyPathToIdeals`: Given a supersingular elliptic curve E_0 and its endomorphism ring $\mathcal{O}_0$, and an isogeny path $E_0 \to E_1$, compute the endomorphism ring $\mathcal{O}_1$ of E_1 and the connecting ideal of $\mathcal{O}_0$ and $\mathcal{O}_1$ (see [GPS19, Section 4.4]).
- `PowersmoothIdeal`: Given a left $\mathcal{O}_0$-ideal I, find another left $\mathcal{O}_0$-ideal J, in the same ideal class as I, such that $N(J)$ is powersmooth. (see [GPS19, Section 4.3], which generalizes the result of [KLPT14]).
- `IdealsToIsogenyPath`: Given $E_0, \mathcal{O}_0$ as above, and E_2, and a $\mathcal{O}_0$-$\mathrm{End}_{\overline{\mathbb{F}_p}}(E_2)$-ideal I, find an isogeny path $E_0 \to E_2$ corresponding to I (see [GPS19, Section 4.5]).

The core of Galbraith's signature scheme is the following zero-knowledge identification protocol which uses the algorithm to the above questions as components:

Algorithm A.1.1 (The identification scheme in [GPS19, Figure 1])**.**
Settings: The prover possesses the private key, which is an isogeny $\psi : E_0 \to E_1$ with $\deg \psi$ being powersmooth. The verifier knows the public key (E_0, E_1).
Goal: The prover proves to the verifer the possession of the private key, and the verifier verifies it.

```
The prover starts a random walk from $E_1$ along the powersmooth isogeny graph and
  generates a powersmooth isogeny $\varphi : E_1 \to E_2$. The prover sends $E_2$ to the verifier.
The verifier generates a random bit $b$ and send it to the prover.
if $b = 0$ then
    The prover sends $\varphi : E_1 \to E_2$ to the verifier
else
    The prover uses algorithm IsogenyPathToIdeals to compute $\mathrm{End}_{\overline{\mathbb{F}_p}}(E_2)$ and the
      connecting ideal $I$ between $\mathrm{End}_{\overline{\mathbb{F}_p}}(E_0)$ and $\mathrm{End}_{\overline{\mathbb{F}_p}}(E_2)$.
    Use the algorithm in [KLPT14] to construct an ideal $J$ in the same left ideal class as
      $I$, with powersmooth norm.
    The prover uses algorithm IdealsToIsogenyPath to compute an alternate isogeny
      $\eta : E_0 \to E_2$ corresponding to the ideal $J$, and send $\eta : E_0 \to E_2$ to the verifier.
end if
The verifier accepts the proof if the returned $\varphi$ is indeed an isogeny $E_1 \to E_2$, or the
  returned $\eta$ is indeed an isogeny $E_0 \to E_2$.
```

2 A generalization to genus 2

Overview of the required routines. We need the following questions to be hard (say, polynomial time in $\log p$):

- `IsogenyPathToMatrix`: Given An isogeny chain $E_0^2 = A_0 \xrightarrow{\phi_1} A_1 \cdots \xrightarrow{\phi_r} A_r$, where each ϕ_i is an (ℓ_i, ℓ_i)-isogeny for some prime ℓ_i. Suppose in addition that $N = \prod_{i-1}^r \ell_i$ is $O(\log p)$-powersmooth. For each $1 \leq i \leq r$, find γ_i and $g_i = \ell_i (\overline{\gamma_i}^t)^{-1} g_{i-1} \gamma_i^{-1}$ (as in Proposition 2.2.9).

- `PowersmoothMatrix`: Given $g, g' \in \mathrm{Mat}_2^+(\mathcal{O})$, find a $O(\log p)$-powersmooth $N = \prod_{i=1}^r \ell_i^{e_i}$ and $\gamma \in \mathrm{Mat}_2(\mathcal{O})$, $\tau, \tau' \in \mathrm{Mat}_2^1(\mathcal{O})$, such that $N(\gamma) = N$ and $N(\overline{\tau}^t g \tau) = \overline{\gamma}^t (\overline{\tau'}^t g' \tau')\gamma$, and γ represents an isogeny path which consists of e_i (ℓ_i, ℓ_i)-isogenies for $1 \leq i \leq r$.

- `MatrixToIsogenyPath`: Given a superspecial abelian surface $A_0/\overline{\mathbb{F}_p}$, a representative $g_0 \in \mathrm{Mat}_2^+(\mathcal{O})$ of A, and $\gamma \in \mathrm{Mat}_2(\mathcal{O})$, where $N = N(\gamma)^{1/2}$ is $O(\log p)$-powersmooth. Find $\gamma_1, \cdots \gamma_r$ and $A_1 \cdots A_r$, such that for each $1 \leq i \leq r$, $N(\gamma_i) = \ell_i^2$ for some prime ℓ_i, $\gamma = \gamma_r \cdots \gamma_1$, and γ_i represents an (ℓ_i, ℓ_i)-isogeny $A_{i-1} \to A_i$.

Here, we give a rough sketch how to set up the identification scheme. This is supposed to be a generalization of Algorithm A.1.1.

Settings for the superspecial signature scheme. We first describe the system parameters. Let p be a prime of the form $p = 4M - 1$, where $M = p_1 \cdots p_s$ is a product of small primes. Choose the base supersingular elliptic curve as $E_0 : y^2 = x^3 + Ax$ for some $A \in \mathbb{F}_{p^2}$ such that $|E_0(\mathbb{F}_p)| = p + 1$ and $|E_0(\mathbb{F}_{p^2})| = (p+1)^2$. In this case, $\mathrm{End}_{\overline{\mathbb{F}_p}}(E_0)$ can be explicitly computed as a maximal order $\mathcal{O} \subseteq B_{p,\infty}$. And when $A_0 := E_0^2$, $\mathrm{End}_{\overline{\mathbb{F}_p}}(A_0)$ can be embedded in $\mathrm{Mat}_2(B_{p,\infty})$.

We first propose the identification scheme, analogous to Algorithm A.1.1. In the identification scheme, there is a prover and a verifier. The prover possesses a private key, which is is an isogeny $\psi : A_0 = E_0^2 \to A_1$, where ψ is a chain of $(2,2)$-isogenies, while the public key is A_0 and A_1.

Algorithm A.2.1 (The genus 2 identification scheme, analogous to algorithm A.1.1 as in [GPS19, Figure 1]).
Settings: The prover possesses the private key, which is an isogeny $\psi : A_0 := E_0^2 \to A_1$, where ψ is a composition of $(2,2)$-isogenies. The prover also precomputes and $\gamma_1 \in \mathrm{Mat}_2^1(\mathcal{O})$ and $g_1 \in \mathrm{Mat}_2^+(\mathcal{O})$ corresponding to A_1 using `IsogenyPathToMatrix`. The verifier knows the public key A_1.
Goal: The prover proves to the verifer the possession of the private key, and the verifier verifies it.

1: The prover starts a random walk from A_1 along the $(2,2)$-isogeny graph and generates an isogeny $\varphi : A_1 \to A_2$. The prover sends A_2 to the verifier.

```
2: The verifier generates a random bit $b$ and send it to the prover.
3: if $b = 0$ then
4:     The prover sends $\varphi : A_1 \to A_2$ to the verifier
5: else
6:     The prover invokes IsogenyPathToMatrix to obtain the matrix $g_2 \in \mathrm{Mat}_2^+(\mathcal{O})$
         corresponding to $A_2$.
7:     The prover invokes PowersmoothMatrix to find $\gamma \in \mathrm{Mat}_2(\mathcal{O})$ and $\tau, \tau' \in \mathrm{Mat}_2^1(\mathcal{O})$,
         such that $N(\gamma) = 3^e$ and $3^e(\overline{\tau}^t\tau) = \overline{\gamma}^t(\overline{\tau'}^t g_2 \tau')\gamma$, and $\gamma$ represents an isogeny path
         $\eta : A_0 \to A_2$ which is a combination of $e$ $(3,3)$-isogenies.
8:     The prover invokes MatrixToIsogenyPath to recover from $\gamma$ the actual $(3,3)$-
         isogeny path $\eta : A_0 \to A_2$. The prover then sends $\eta : A_0 \to A_2$ to the verifier.
9: end if
10: The verifier accepts the proof if the returned $\varphi$ is indeed an isogeny $A_1 \to A_2$, or the
    returned $\eta$ is indeed an isogeny $A_0 \to A_2$.
```

In particular, we will give a proposed algorithm for the `IsogenyToMatrix` problem, which showed up in step 6 in Algorithm A.2.1.

Algorithm A.2.2 (`IsogenyPathToMatrix`)**.**
Input: A supersingular elliptic curve E_0 defined over $\mathbb{F}_{p^2}$, $\mathcal{O} = \mathrm{End}_{\overline{\mathbb{F}_p}}(E_0) = \mathbb{Z}\psi_1 + \mathbb{Z}\psi_2 + \mathbb{Z}\psi_3 + \mathbb{Z}\psi_4$, $B = \mathcal{O} \otimes_{\mathbb{Z}} \mathbb{Q}$, a chain of isogenies $A_0 = E_0^2 \xrightarrow{\phi_1} A_1 \to \cdots \xrightarrow{\phi_r} A_r$, where each ϕ_i is a (ℓ_i, ℓ_i)-isogeny specified by the kernel $K_i := \ker(\phi_{i-1})$.
Output: For each $1 \leq i \leq r$, the matrices $\gamma_i \in \mathrm{Mat}_2(\mathcal{O})$ and $g_i \in \mathrm{Mat}_2^+(\mathcal{O})$, as described in Proposition 2.2.9.

```
1: Compute $N = \prod_{i=1}^r \ell_i$, and let $N = \prod_{j=1}^s p_j^{e_j}$ be the prime factorization (so $\ell_i \in \{p_j\}_{j=1}^s$).
2: for $j = 1, \cdots, s$ do
3:     Compute a basis $\{P_{j,1}, P_{j,2}\}$ of $E_0[p_j^{e_j}]$.
4:     Compute the Weil pairing matrix $W_j \in \mathrm{Mat}_4(\mathbb{Z}/p_j\mathbb{Z})$ with respect to the basis
         $\{(p_i^{e_i-1}P_{j,1}, O), (p_i^{e_i-1}P_{j,2}, O), (O, p_i^{e_i-1}P_{j,1}, (O, p_i^{e_i-1}P_{j,2}))\}$ and the product
         polarization.
5:     Compute $\delta_{j,1}, \delta_{j,2}, \delta_{j,3}, \delta_{j,4} \in \mathrm{Mat}_2(\mathbb{Z}/p_j\mathbb{Z})$, the action of $\psi_1, \psi_2, \psi_3, \psi_4$ on $E_0[p_j]$ with
         respect to the basis $\{p_j^{e_j-1}P_{j,1}, p_j^{e_j-1}P_{j,2}\}$.
6: end for
7: Set $\phi = id_{E_0^2}$ and $g = \gamma = id \in \mathrm{Mat}_2(\mathcal{O})$.
8: for $i = 1, \cdots, r$ do
9:     Find a basis $\{S_{i,1}, S_{i,2}\}$ of $K_i$.
10:    Suppose $\ell_i = p_j$. Solve DLP to get $u_1, u_2, u_3, u_4$ and $v_1, v_2, v_3, v_4$ satisfying
```

```
          $S_{i,1} = u_1\phi((P_{j,1},O)) + u_2\phi((P_{j,2},O)) + u_3\phi((O,P_{j,1})) + u_4\phi((O,P_{j,2}))$ and
          $S_{i,2} = v_1\phi((P_{j,1},O)) + v_2\phi((P_{j,2},O)) + v_3\phi((O,P_{j,1})) + v_4\phi((O,P_{j,2}))$.
11:     Compute $T_{i,1} = \gamma(u_1(P_{j,1},O) + u_2(P_{j,2},O) + u_3(O,P_{j,1}) + u_4(O,P_{j,2}))$ and
          $T_{i,2} = \gamma(v_1(P_{j,1},O) + v_2(P_{j,2},O) + v_3(O,P_{j,1}) + v_4(O,P_{j,2}))$.
12:     Compute $\Gamma \in \mathrm{Mat}_4(\mathbb{Z}/p_j\mathbb{Z})$ from $g$, and compute $W = W_j\Gamma$.
                                                        ▷ Use the mapping in line 5.
13:     if $[u_1\ u_2\ u_3\ u_4]W[v_1\ v_2\ v_3\ v_4]^t \neq 0$ then
14:         return error.                               ▷ Implies the kernel is not isotropic.
15:     end if
16:     List $V = \{\Lambda \in \mathrm{Mat}_4(\mathbb{Z}/p_j\mathbb{Z}) \mid \mathrm{rank}(\Lambda) = 2, \Lambda[u_1\ u_2\ u_3\ u_4]^t = \Lambda[v_1\ v_2\ v_3\ v_4]^t = 0\}$.
17:     Find $\Lambda_i \in V$ and $\gamma_i \in \mathrm{Mat}_2(\mathcal{O})$ such that $\gamma_i$ acts as $\Lambda_i$ on $E_0^2[p_j]$ and $N(\gamma_i) = p_j^2$.
                                                        ▷ If possible, choose $\Lambda_i$ and $\gamma_i$ such that $\overline{\gamma_i}^t = \gamma_i$.
18:     Compute $g_i = p_j(\overline{\gamma_i}^t)^{-1}g\gamma_i^{-1}$, replace $g$ by $g_i$, and replace $\gamma$ by $\gamma_i\gamma$.
19: end for
20: return $\{\gamma_i, g_i\}_{i=1}^r$.
```

For what is relevant to Chapter 2, step 17 can be solved if we can find a principal ideal generator of $\Gamma\mathrm{Mat}_2(\mathcal{O}) + p_j\mathrm{Mat}_2(\mathcal{O})$.

Bibliography

[ALV04] A. Agashe, K. Lauter, and R. Venkatesan. *Constructing elliptic curves with a known number of points over a prime field*, volume 41, pages 1–18. Fields Institute Communications, 2004.

[BFS15] Magali Bardet, Jean-Charles Faugère, and Bruno Salvy. On the complexity of the F5 Gröbner basis algorithm. *Journal of Symbolic Computation*, 70:49–70, September 2015.

[BGL11] Reinier Bröker, David Gruenewald, and Kristin Lauter. Explicit CM theory for level 2-structures on abelian surfaces. *Algebra & Number Theory*, 5(4):495 – 528, 2011.

[Bis15] Gaetan Bisson. Computing endomorphism rings of abelian varieties of dimension two. *Mathematics of Computation*, 84:1977–1989, 01 2015.

[BJW17] E. H. Brooks, Dimitar Jetchev, and B. Wesolowski. Isogeny graphs of ordinary abelian varieties. *Research in Number Theory*, 3(28):1–38, 2017.

[Bro93] Bradley Brock. *Superspecial curves of genera two and three*. PhD thesis, Princeton University, 1993.

[Brö08] Reinier Bröker. A p-adic algorithm to compute the Hilbert class polynomial. *Mathematics of Computation*, 77(264):2417–2435, 2008.

[Buc88] Johannes Buchmann. A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. *Séminaire de théorie des nombres, Paris*, 1989(1990):27–41, 1988.

[Car10] Robert Carls. Fast point counting on genus two curves in characteristic three, 2010.

[CDS20] Wouter Castryck, Thomas Decru, and Benjamin Smith. Hash functions from superspecial genus-2 curves using richelot isogenies. *Journal of Mathematical Cryptology*, 14, 08 2020.

[CGL08] Denis Charles, Eyal Goren, and Kristin Lauter. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22:93–113, 12 2008.

[CKL08] Robert Carls, David Kohel, and David Lubicz. Higher-dimensional 3-adic CM construction. *Journal of Algebra*, 319(3):971–1006, 2008.

[CKM97] S. Collart, M. Kalkbrener, and D. Mall. Converting bases with the gröbner walk. *Journal of Symbolic Computation*, 24:465–469, 1997.

[CL09] Robert Carls and David Lubicz. A p-Adic Quasi-Quadratic Time Point Counting Algorithm. *International Mathematics Research Notices*, 2009(4):698–735, 01 2009.

[CO12] Ching-Li Chai and Frans Oort. Abelian varieties isogenous to a Jacobian. *Annals of Mathematics*, 176(1):589–635, 2012.

[Coh13] Henri Cohen. *A course in computational algebraic number theory*. Graduate Texts in Mathematics. Springer Berlin Heidelberg, 2013.

[CR15] Romain Cosset and Damien Robert. Computing (l, l)-isogenies in polynomial time on Jacobians of genus 2 curves. *Mathematics of Computation*, 84(294):1953–1975, 2015. Accepté pour publication à Mathematics of Computations.

[DJP14] Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.

[EHL$^+$18] Kirsten Eisenträger, Sean Hallgren, Kristin Lauter, Travis Morrison, and Christophe Petit. Supersingular isogeny graphs and endomorphism rings: Reductions and solutions. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018*, pages 329–368, Cham, 2018. Springer International Publishing.

[EL10] Kirsten Eisentraerger and Kristin Lauter. A CRT algorithm for constructing genus 2 curves over finite fields. In *Arithmetics, geometry, and coding theory (AGCT 2005)*, volume 21 of *Sémin. Congr.*, pages 161–176, Soc. Math. France, Paris, 2010.

[Eng09] Andreas Enge. The complexity of class polynomial computation via floating point approximations. *Mathematics of Computation*, 78(266):1089–1107, 2009.

[ET14] Andreas Enge and Emmanuel Thomé. Computing class polynomials for abelian surfaces. *Experimental Mathematics*, 23(2):129–145, 2014.

[Fau99] Jean-Charles Faugére. A new efficient algorithm for computing Gröbner bases (f4). *Journal of Pure and Applied Algebra*, 139(1):61–88, 1999.

[FGLM93] J.C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional gröbner bases by change of ordering. *Journal of Symbolic Computation*, 16(4):329–344, 1993.

[FL08] David Freeman and Kristin Lauter. *Computing endomorphism rings of Jacobians of genus 2 curves over finite fields*, pages 29–66. World Scientific, 2008.

[FLR11] Jean-Charles Faugére, David Lubicz, and Damien Robert. Computing modular correspondences for abelian varieties. *Journal of Algebra*, 343(1):248–277, 2011.

[FM17] Jean-Charles Faugére and Chenqi Mou. Sparse FGLM Algorithms. *Journal of Symbolic Computation*, 80:538–569, 2017.

[FT19] E. V. Flynn and Yan Bo Ti. Genus two isogeny cryptography. In Jintai Ding and Rainer Steinwandt, editors, *Post-Quantum Cryptography*, pages 286–306, Cham, 2019. Springer International Publishing.

[GHK$^+$06] P. Gaudry, T. Houtmann, D. Kohel, C. Ritzenthaler, and A. Weng. The 2-Adic CM Method for Genus 2 Curves with Application to Cryptography. In Xuejia Lai and Kefei Chen, editors, *Advances in Cryptology – ASIACRYPT 2006*, pages 114–129, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[GMP19] Zhenchao Ge, Micah B. Milinovich, and Paul Pollack. A note on the least prime that splits completely in a nonabelian galois number field. *Mathematische Zeitschrift*, 292:183–192, 2019.

[Gor97] Eyal Z. Goren. On certain reduction problems concerning abelian surfaces. *Manuscripta Mathematica*, 94:33–43, 1997.

[GPS19] Steven Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. *Journal of Cryptology*, 33, 03 2019.

[Gra08] A. Granville. Smooth numbers: computational number theory and beyond. In *Algorithmic Number Theory: Lattices, Number Fields, Curves, and Cryptography*, volume 44, pages 267–323, 2008.

[HJ20] Tommy Hofmann and Henri Johnston. Computing isomorphisms between lattices. *Mathematics of Computation*, 89(326):2931–2963, November 2020.

[HM06] Emmanuel Hallouin and Christian Maire. Cancellation in totally definite quaternion algebras. *Journal Fur Die Reine Und Angewandte Mathematik*, 2006:189–213, 06 2006.

[Ibu19] Tomoyoshi Ibukiyama. Quinary lattices and binary quaternion hermitian lattices. *Tohoku Mathematical Journal*, 71(2):207–220, 2019.

[Igu60] Jun-Ichi Igusa. Arithmetic variety of moduli for genus two. *Annals of Mathematics*, 72(3):612–649, 1960.

[IKO86] Tomoyoshi Ibukiyama, Toshiyuki Katsura, and Frans Oort. Supersingular curves of genus two and class numbers. *Compositio Mathematica*, 57(2):127–152, 1986.

[JW15] Dimitar Jetchev and Benjamin Wesolowski. Horizontal isogeny graphs of ordinary abelian varieties and the discrete logarithm problem. https://arxiv.org/abs/1506.00522, 2015.

[KL14] Ming Hsuan Kang and Wen Ching Winnie Li. Zeta functions of complexes arising from $\mathrm{PGL}(3)$. *Advances in Mathematics*, 256:46–103, May 2014.

[KLPT14] David Kohel, Kristin Lauter, Christophe Petit, and Jean-Pierre Tignol. On the quaternion -isogeny path problem. *LMS Journal of Computation and Mathematics*, 17, 06 2014.

[Koh] David R. Kohel. Echidna algorithms: Algorithms for elliptic curves and higher dimensional analogues. http://iml.univ-mrs.fr/ kohel/alg/index.html.

[Koh96] David Kohel. *Endomorphism Rings of Elliptic Curves over Finite Fields*. PhD thesis, University of California, Berkley, 1996.

[Koh08] David R. Kohel. Complex multiplication and canonical lifts. In *Algebraic Geometry and Its Applications*, pages 67–83. World Scientific, 2008.

[KV10] Markus Kirschmer and John Voight. Algorithmic enumeration of ideal classes for quaternion orders. *SIAM Journal on Computing*, 39(5):1714–1747, 2010.

[Lan87] Serge Lang. *Elliptic Functions*, volume 112 of *Graduate Texts in Mathematics*. Springer-Verlag New York, 1987.

[Le94] M. Le. Upper bounds for class numbers of real quadratic fields. *Acta Arithmetica*, 68(2):141–144, 1994.

[LL06] Reynald Lercier and David Lubicz. A quasi quadratic time algorithm for hyperelliptic curve point counting. *The Ramanujan Journal*, 12:399–423, 2006.

[LO77] J. C. Lagarias and A. M. Odlyzko. Effective versions of the Chebotarev density theorem. In A. Fröhlich, editor, *Algebraic Number Fields: L-functions and Galois Properties*, pages 409–464. Academic Press, London, 1977.

[LO98] Ke-Zheng Li and Frans Oort. *Moduli of supersingular abelian varieties*, volume 1680. Springer Science & Business Media, 1998.

[Lou03] Stephane Louboutin. Explicit lower bounds for residues at $s = 1$ of Dedekind zeta functions and relative class numbers of CM-fields. *Transactions of the American Mathematical Society*, 355(08):3079–3098, 2003.

[LST64] J. Lubin, J-P. Serre, and J. Tate. Elliptic curves and formal groups. Lecture notes prepared in connection with the seminars held at the Summer Institute of Algebraic Geometry, Whitney Estate, Woods Hole, Massachusetts, July 6-July 31, 1964.

[LV15] Kristin Lauter and Bianca Viray. An arithmetic intersection formula for denominators of Igusa class polynomials. *American Journal of Mathematics*, 137(2):497–533, 2015.

[LY11] Kristin Lauter and Tonghai Yang. Computing genus 2 curves from invariants on the Hilbert moduli space. *Journal of Number Theory*, 131(5):936–958, 2011. Elliptic Curve Cryptography.

[Mes72] William Messing. *The Crystals Associated to Barsotti-Tate Groups: With Applications to Abelian Schemes*, volume 264 of *Lecture Notes in Mathematics*. Springer-Verlag Berlin Heidelberg, 1972.

[Mes91] Jean-François Mestre. *Construction de courbes de genre 2 à partir de leurs modules*, pages 313–334. Birkhäuser Boston, Boston, MA, 1991.

[Mil08] J.S. Milne. Abelian varieties. `http://www.jmilne.org/math/CourseNotes/av.html`, 2008.

[Mum66] David Mumford. On the equations defining abelian varieties. I. *Inventiones Mathematicae*, 1(4):287–354, 1966.

[Mum67] David Mumford. On the equations defining abelian varieties. II. *Inventiones Mathematicae*, 3(2):75–135, 1967.

[Mum06] David Mumford. *Tata Lectures on Theta II: Jacobian theta functions and differential equations*. Modern Birkhäuser Classics. Birkhäuser, 2006.

[Nat16] National Institute of Standards and Technology. Post-quantum cryptography. `https://csrc.nist.gov/projects/post-quantum-cryptography`, 2016.

[OU73] F. Oort and K. Ueno. Principally polarized abelian variaties dimension two or three are Jacobian varieties. *Journal of the Faculty of Science. University of Tokyo. Section IA. Mathematics*, 20(3):377–381, 1973.

[Pag14] Aurel Page. An algorithm for the principal ideal problem in indefinite quaternion algebras. In *Algorithmic Number Theory Symposium ANTS XI*, volume 17 of *LMS Journal of Computation and Mathematics*, pages 366–384, GyeongJu, South Korea, August 2014.

[Rei03] Irving Reiner. *Maximal Orders*. London Mathematical Society monographs series: London Mathematical Society. Clarendon Press, 2003.

[RL13] Damien Robert and Kristin Lauter. Improved CRT algorithm for class polynomials in genus 2. In *Tenth Algorithmic Number Theory Symposium*, volume 1 of *The Open Book Series*, pages 437–461, 2013.

[Shi79] Tetsuji Shioda. Supersingular $k3$ surfaces. In *Algebraic geometry*, pages 564–591. Springer, 1979.

[Shi98] Goro Shimura. *Abelian Varieties with Complex Multiplication and Modular Functions*. Princeton University Press, 1998.

[Sil94] Joseph Silverman. *Advanced Topics in the Arithmetic of Elliptic Curves*, volume 151 of *Graduate Texts in Mathematics*. Springer, New York, NY, 1994.

[Spa94] Anne-Monika Spallek. *Kurven vom Geschlecht 2 und ihre Anwendung in Public-Key-Kryptosystemen*. PhD thesis, Institut für Experimentelle Mathematik (Essen), 1994.

[Spr19] Caleb Springer. Computing the endomorphism ring of an ordinary abelian surface over a finite field. *Journal of Number Theory*, 202:430–457, 2019.

[Str10] Marco Streng. *Complex Multiplication of Abelian Surfaces*. PhD thesis, Universiteit Leiden, 2010.

[Str14] Marco Streng. Computing igusa class polynomials. *Mathematics of Computation*, 83(285):275–309, 2014.

[Tak18] Katsuyuki Takashima. Efficient algorithms for isogeny sequences and their cryptographic applications. In *Mathematical Modelling for Next-Generation Cryptography: CREST Crypto-Math Project*, pages 97–114. Springer Singapore, Singapore, 2018.

[Voi20] John Voight. *Quaternion Algebras*. Graduate Texts in Mathematics. Springer International Publishing, 2020.

[vW99] Paul B. van Wamelen. Examples of genus two CM curves defined over the rationals. *Mathematics of Computation*, 68(225):307–320, 1999.

[Wat69] William C. Waterhouse. Abelian varieties over finite fields. *Annales scientifiques de l'École Normale Supérieure*, Ser. 4, 2(4):521–560, 1969.

[YY09] Chia-Fu Yu and Jeng-Daw Yu. Mass formula for supersingular abelian surfaces. *Journal of Algebra*, 322(10):3733–3743, 2009.

Vita

Hao-Wei Chu

The author was born 1986 in Kaohsiung, Taiwan. He went to National Taiwan University and obtained B.S. in Electrical Engineering and Mathematics in 2006. He then entered the Graduate Institute of Communication Engineering in National Taiwan University with a Master degree in 2008. After working for a while, he decided to return to mathematics and earned a Master degree in Mathematics in National Taiwan University in 2013 under the supervision of Professor Jing Yu. He then got an opportunity to study as a research trainee in Institute of Mathematics, Academia Sinica in Taiwan, under the supervision of Professor Chia-Fu Yu. He then enrolled in the mathematics Ph. D. program at Penn State in 2015 and studied number theory, arithmetic geometry, and cryptography under the supervision of Professor Kirsten Eisenträger.