Computing Conjectural Orders of Shafarevich-Tate Groups of Modular Abelian Varieties

William Stein

June 3, 2009

Introduction

This paper is motivated by the following open problem.

Problem. Let $A = A_f$ be a modular abelian variety attached to a newform $f \in S_2(\Gamma_0(N))$, and assume that $L(A,1) \neq 0$ so that A has rank 0 (by [**?**]). Compute

$$Ш(A)_{\mathrm{an}} = \frac{L(A,1) \cdot \#A(\mathbb{Q})_{\mathrm{tor}} \cdot \#A^\vee(\mathbb{Q})_{\mathrm{tor}}}{\Omega_A \cdot \prod c_p}.$$

The author and others have given numerous partial results toward this problem in [] (see my Magma BSD paper), but the general problem remains open. In particular, there is no known general algorithm to compute the rational number $L(A,1)/\Omega_A$ in all cases, no known way to compute c_p in general, nor any known way to compute $\#A(\mathbb{Q})_{\mathrm{tor}}$ or $\#A^\vee(\mathbb{Q})_{\mathrm{tor}}$. However, there are algorithms to compute a divisor and multiple of each of these quantities, hence a divisor and multiple of $Ш(A)_{\mathrm{an}}$.

Here we explain algorithms for computing or bounding the quantities appearing in the above formula, and emphasize precisely what is *not* known. We assume the reader has a background in algebraic number theory and elliptic curves.

Contents

1 The Conjectural Order of the Shafarevich-Tate Group

1.1 The Basic Objects

Let $f \in S_2(\Gamma_0(N))$ be a weight 2 newform on $\Gamma_0(N)$. Thus $f = \sum_{n=1}^{\infty} a_n q^n$, with $q = e^{2\pi i z}$ is a holomorphic function on the upper half plane such that $f(z)dz$ is invariant under the action of $\Gamma_0(N)$. Also, f is normalized so that $a_1 = 1$, and f is *new* in the sense that f is not in the sum of the natural inclusions $f(q) \mapsto f(q)$ and $f(q) \mapsto f(q^p)$

$$S_2(\Gamma_0(N/p)) \hookrightarrow S_2(\Gamma_0(N))$$

for all $p \mid N$ prime. Finally, f is an eigenvector for all the Hecke operators T_p, where

$$T_p\left(\sum a_n q^n\right) = \sum_{m \in \mathbb{Z}} (a_{mp} + p a_{m/p}) q^m$$

and $a_{m/p} = 0$ if $m/p \notin \mathbb{Z}$, and we omit the $pa_{m/p}$ term if $p \mid N$. From the above formulas we see that since f is an eigenvector, we must have $T_p(f) = a_p f$, i.e., the eigenvalue of T_p is a_p. There is also a defintion of Hecke operators T_n for any positive integer n, and $T_n(f) = a_n f$.

Example 1.1. We make a table of all weight 2 newforms for $N \leq 30$:

```
sage: for N in [1..30]:
...       S = CuspForms(N).newforms('a')
...       if len(S)>0: print N, S
11 [q - 2*q^2 - q^3 + 2*q^4 + q^5 + O(q^6)]
14 [q - q^2 - 2*q^3 + q^4 + O(q^6)]
15 [q - q^2 - q^3 - q^4 + q^5 + O(q^6)]
17 [q - q^2 - q^4 - 2*q^5 + O(q^6)]
19 [q - 2*q^3 - 2*q^4 + 3*q^5 + O(q^6)]
20 [q - 2*q^3 - q^5 + O(q^6)]
21 [q - q^2 + q^3 - q^4 - 2*q^5 + O(q^6)]
23 [q + a0*q^2 + (-2*a0 - 1)*q^3 + (-a0 - 1)*q^4 + 2*a0*q^5 + O(q^6)]
24 [q - q^3 - 2*q^5 + O(q^6)]
26 [q - q^2 + q^3 + q^4 - 3*q^5 + O(q^6),
    q + q^2 - 3*q^3 + q^4 - q^5 + O(q^6)]
27 [q - 2*q^4 + O(q^6)]
29 [q + a0*q^2 - a0*q^3 + (-2*a0 - 1)*q^4 - q^5 + O(q^6)]
30 [q - q^2 + q^3 + q^4 - q^5 + O(q^6)]
```

We also compute the fields defined by a_0 above:

```
sage: f = CuspForms(23).newforms('a')[0]
sage: f.hecke_eigenvalue_field()
Number Field in a0 with defining polynomial x^2 + x - 1
sage: CuspForms(29).newforms('a')[0].hecke_eigenvalue_field()
Number Field in a0 with defining polynomial x^2 + 2*x - 1
```

The ring $\mathbb{T} = \mathbb{Z}[T_1, T_2, T_3, \dots]$ generated by all the Hecke operators T_n is a commutative ring that is finite as a $\mathbb{Z}$-module. In fact, when N is cube-free, there is a list of totally real (all embeddings are real) numbers fields $K_1, \dots, K_m$ such that

$$\mathbb{T} \hookrightarrow \mathcal{O}_{K_1} \times \cdots \times \mathcal{O}_{K_m},$$

where the $\mathcal{O}_{K_i}$ are the rings of integers of the K_i. When a cube divides N, then $\mathbb{T}$ usually contains nilpotents, so can't embed in a product of fields (which contains no nilpotents).

We associate to our newform $f = \sum a_n q^n$ an ideal I_f in $\mathbb{T}$ as follows:

$$I_f = \mathrm{Ann}_{\mathbb{T}}(f) = \{t \in \mathbb{T} : tf = 0\}.$$

Let $\mathcal{O}_f = \mathbb{Z}[a_1, a_2, a_3, \dots]$ be the ring generated by the Fourier coefficient a_n of f, and let $K_f = \mathrm{Frac}(\mathcal{O}_f)$ be its fraction field, which is a number field. We have a surjective homomoprhism

$$\mathbb{T} \to \mathcal{O}_f, \qquad T_n \mapsto a_n$$

and an exact sequence

$$0 \to I_f \to \mathbb{T} \to \mathcal{O}_f \to 0.$$

The ring $\mathcal{O}_f$ is an order in the ring of integers of a number field.

Example 1.2. For the forms of level 23 and 29 above, $\mathcal{O}_f$ has discriminant 5 and 8, respectively. In both cases $\mathcal{O}_f$ is the maximal order in a real quadratic field. The first example where $\mathcal{O}_f$ is not maximal occurs at level $N = 69$, where the order is $\mathbb{Z}[\sqrt{5}]$.

```
sage: f = CuspForms(69).newforms('a')[1]
sage: f.hecke_eigenvalue_field().order([f[2]])
Order in Number Field in a1 with defining polynomial x^2 - 5
```

The extended upper half plane is

$$\mathfrak{h}^* = \mathfrak{h} \cup \mathbb{Q} \cup \{i\infty\} = \mathfrak{h} \cup \mathbb{P}^1(\mathbb{Q}).$$

We can view $f(z)dz$ as a holomorphic differential on the *modular curve*

$$X_0(N)(\mathbb{C}) = \Gamma_0(N) \backslash \mathfrak{h}^*.$$

The modular curve $X_0(N)$ is an algebraic curve of genus $g = \dim S_2(\Gamma_0(N))$ with a canonical model over $\mathbb{Q}$. It's $\overline{\mathbb{Q}}$ affine points are in bijection with the isomorphism classes of pairs (E, C) where E is an elliptic curve and $C \subset E$ is a cyclic subgroup of order N. The Hecke algebra $\mathbb{T}$ also acts via correspondences defined over $\mathbb{Q}$ on the group $\mathrm{Div}(X_0(N))$ of divisors on $X_0(N)$. This induces an action of $\mathbb{T}$ on the Jacobian $J_0(N) = \mathrm{Jac}(X_0(N))$. The Jacobian $J_0(N)$ is a projective algebraic variety of dimension g defined over $\mathbb{Q}$ whose K points functorially parametrize the group $\mathrm{Pic}^0(X_0(N)/K)$ of divisor classes of degree 0

on $X_0(N)$ that are rational over a field K. In particular, $J_0(N)$ has an algebraic group structure, so is an abelian variety.

We are now in a position to associate the abelian variety A_f to f, following a construction introduced first by Shimura. Let

$$A_f = J_0(N)/I_f J_0(N).$$

This is an abelian variety because quotients of abelian varieties by abelian varieties are abelian varieties, and $I_f J_0(N)$, which is the sum of the images of $J_0(N)$ under all elements of I_f, is an abelian subvariety of $J_0(N)$. Since $\mathbb{T}$ is commutative, $I_f J_0(N)$ is preserved by $\mathbb{T}$, so $\mathbb{T}$ acts on A_f. Moreover, I_f acts as 0 on A_f, so $\mathcal{O}_f = \mathbb{T}/I_f$ also acts on A_f. The abelian variety A_f is simple over $\mathbb{Q}$, meaning it is not isogenous to a product $B \times C$ of nonzero abelian varieties, and A_f has dimension equal to the degree of the field K_f generated by the a_n.

A $\Gamma_0(N)$-modular abelian variety A is any abelian variety quotient of $J_0(N)$. The abelian varieities A_f are primes in the context of modular abelian varieties, since every modular abelian variety A is isogenous to a product of the A_f's. It is also a recent deep theorem that if A is any simple abelian variety over $\mathbb{Q}$ with endomorphism ring an order in a totally real number field of degree $\dim(A)$, then A is isogenous to some A_f.

Let $f_1, \ldots, f_d$ be the Galois conjugates of $f = \sum a_n q^n$, i.e., the orbert of the $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ action on q-expansions in $\overline{\mathbb{Q}}[[q]]$ via the action on coefficients. The L-function $L(A_f, s)$ is equal to

$$L(A_f, s) = \prod_{i=1}^{d} L(f_i, s),$$

where for each i, the L-function of $f_i = \sum a_n^{(i)} q^n$ is

$$L(f_i, s) = \sum_{n=1}^{\infty} \frac{a_n^{(i)}}{n^s}.$$

Hecke proved that $L(f_i, s)$ extends to a holomorphic function on all $\mathbb{C}$.

The Birch and Swinnerton-Dyer conjecture relates the order of vanishing of $L(A_f, s)$ to the rank of the Mordell-Weil group $A_f(\mathbb{Q})$ of rational points on A_f.

Conjecture 1.3 (Birch and Swinnerton-Dyer Rank Conjecture)**.** *We have*

$$\mathrm{ord}_{s=1} L(A_f, s) = \mathrm{rank}(A_f(\mathbb{Q})).$$

The Shafarevich-Tate group of A_f is

$$Ш(A_f) = \ker\left(\mathbb{H}^1(\mathbb{Q}, A) \to \bigoplus_{v \le \infty} \mathbb{H}^1(\mathbb{Q}_v, A) \right).$$

There is also a conjecture about the leading coefficient about $s = 1$ of the Taylor expansion of $L(A_f, s)$. In the special case when $L(A_f, 1) \neq 0$, this takes the following form.

Conjecture 1.4 (Birch and Swinnerton-Dyer Formula (Rank 0))**.** *Let $A = A_f$ and assume that $L(A,1) \neq 0$. Then*

$$\#Ш(A) = \frac{L(A,1) \cdot \#A(\mathbb{Q})_{\mathrm{tor}} \cdot \#A^\vee(\mathbb{Q})_{\mathrm{tor}}}{\Omega_A \cdot \prod c_p}. \tag{1.1}$$

We will refer to Conjecture 1.4 as the "BSD conjecture" in the rest of this paper.

The goal of this paper is to explain everything known about computing the right hand side of (1.1), and hopefully inspire further progress on the following open problem: *give a (practical) algorithm to compute the right hand side of* (1.1). The motivation for this problem is emphatically *not* computational complexity or computability. It is to provide further tools that will support gathering data that can be used to improve our theoretical understanding of modular forms and abelian varieties.

1.2 Strategy for computing $Ш(A)_{\mathrm{an}}$

We now outline the basic strategy for computing the right hand side of (1.1).

1.2.1 The L-ratio

Using modular symbols, we can compute the rational number $c_A \cdot L(A,1)/\Omega_A$, where c_A is the Manin constant of A. The Manin constant c_A is an integer that we will revisit below. Modular symbols provide an explicit presentation for the homology group $\mathbb{H}_1(X_0(N), \mathbb{Z}) \approx \mathbb{Z}^{2g}$, along with an action of the Hecke algebra $\mathbb{T}$ on this homology group. Also, Manin proved that if $\alpha, \beta \in \mathbb{P}^1(\mathbb{Q})$, then the real homology class of the path $\{\alpha, \beta\}$ from α to β defines an element of $\mathbb{H}_1(X_0(N), \mathbb{Q})$. Let

$$\Phi : \mathbb{H}_1(X_0(N), \mathbb{Q}) \cong \mathbb{H}_1(J_0(N), A) \to \mathbb{H}_1(A, \mathbb{Q})$$

be the map on homology induced by the $J_0(N) \to A$. There is also an action of complex conjugation on $\mathbb{H}_1(X_0(N), \mathbb{Q})$, and we denote by $\mathbb{H}_1^+$ the $+1$ eigenspace. Also, let

$$c_\infty = \#(A(\mathbb{R})/A(\mathbb{R})^0)$$

where $A(\mathbb{R})^0$ is the connected component containing the identity. There is a simple algorithm to compute c_∞ in terms of the matrix of complex conjugation acting on $\mathbb{H}_1(X_0(N), \mathbb{Z})$. Finally, we have

Theorem 1.5 (Agashe-Stein)**.**

$$c_\infty \cdot c_A \cdot \frac{L(A,1)}{\Omega_A} = [\Phi(\mathbb{H}_1^+) : \Phi(\mathbb{T}\{0,\infty\})],$$

where the index on the right is a lattice index, i.e., after choosing a basis for $\Phi(\mathbb{H}_1^+)$, it is the absolute value of the determinant of any invertible matrix that sends $\Phi(\mathbb{H}_1^+)$ isomorphically onto $\Phi(\mathbb{T}\{0,\infty\})$, or 0 if $\Phi(\mathbb{T}\{0,\infty\}) = 0$.

1.2.2 The Manin constant

Definition 1.6. The *Manin constant* of A is

$$c_A = \# \left(\frac{S_2(\Gamma_0(N), \mathbb{Z})[I_f]}{H^0(\mathcal{A}, \Omega_{\mathcal{A}/\mathbb{Z}})} \right) \in \mathbb{Z},$$

where $\mathcal{A}$ is the Néron model of A.

Conjecture 1.7 (Agashe-Stein)**.** $c_A = 1$

There are strong bounds on the possibilities for c_A, e.g., if $\ell \mid c_A$ is prime, then $\ell^2 \mid 4N$, where $d = \dim(A)$. There is still no known algorithm for computing c_A in general.

1.2.3 Torsion

Next, we use that for any prime $p \nmid N$, if $\chi_p(X)$ is the characteristic polynomial of the Hecke operator T_p acting on $A = A_f$, then

$$\#A(\mathbb{F}_p) = \#A^\vee(\mathbb{F}_p) = \chi_p(p+1),$$

and for $p \nmid 2N$ we have $A(\mathbb{Q})_{\mathrm{tor}} \hookrightarrow A(\mathbb{F}_p)$. Thus the greatest common divisor of the numbers $\chi_p(p+1)$, for $p \nmid 2N$ is a multiple of both $\#A(\mathbb{Q})_{\mathrm{tor}}$ and $\#A^\vee(\mathbb{Q})_{\mathrm{tor}}$. This yields a multiple of $\#A(\mathbb{Q})_{\mathrm{tor}} \cdot \#A^\vee(\mathbb{Q})_{\mathrm{tor}}$.

To obtain a divisor of this product, we consider the subgroup $C \subset J_0(N)$ generated by differences of cusps:

$$C = \langle (\alpha) - (\beta) : \alpha, \beta \in \mathbb{Q} \cup \{\infty\} \rangle.$$

It is possible to explicitly represent and compute with C by using that

$$A(\overline{\mathbb{Q}})_{\mathrm{tor}} \cong \mathbb{H}^1(A, \mathbb{Q})/\mathbb{H}^1(A, \mathbb{Z}).$$

We have $C \subset J_0(N)(\mathbb{Q}(\zeta_N))_{\mathrm{tor}}$, and there is an explicit description of the action of $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ on C, so we can compute the subgroup

$$C \cap A^\vee(\mathbb{Q})_{\mathrm{tor}} \subset J_0(N)(\mathbb{Q})_{\mathrm{tor}}$$

and also the subgroup

$$(\pi_A(C))(\mathbb{Q}) \subset A(\mathbb{Q})_{\mathrm{tor}},$$

where $\pi_A : J_0(N) \to A$ is the natural quotient morphism. We thus obtain subgroups of both $A(\mathbb{Q})_{\mathrm{tor}}$ and $A^\vee(\mathbb{Q})_{\mathrm{tor}}$, hence a divisor of $\#A(\mathbb{Q})_{\mathrm{tor}} \cdot \#A^\vee(\mathbb{Q})_{\mathrm{tor}}$.

1.2.4 Tamagawa Numbers

Let $\mathcal{A}$ be the Néron model of A over $\mathbb{Z}$. This is a smooth commutative group scheme over $\mathbb{Z}$ with the property that for every smooth scheme X over $\mathbb{Z}$ the natural map

$$\mathrm{Hom}(X, \mathcal{A}) \to \mathrm{Hom}(X_{\mathbb{Q}}, A)$$

is an isomorphism. The Néron model $\mathcal{A}$ is unique, up to a unique isomorphism, and it is a deep theorem that $\mathcal{A}$ exists.

Since Néron models are fairly abstract, to work with Néron models, all you really need to know is that they exist and that they have a few functorials properties. For example, if $A \to B$ is a morphism of abelian varieties over $\mathbb{Q}$, then there is an induced map $\mathcal{A} \to \mathcal{B}$ on Néron models, i.e., the association $A \mapsto \mathcal{A}$ is a functor from the category of abelian varieties over $\mathbb{Q}$ to the category of smooth schemes over $\mathbb{Z}$. Néron models over $\mathbb{Z}$ are always smooth (over $\mathbb{Z}$), but they are never proper ("=complete") over $\mathbb{Z}$. I imagine the Néron model $\mathcal{A}$ as A together with a "good" choice of not-necessarily connected reduction "of A" modulo every prime p, glued together to form a group scheme over $\mathbb{Z}$.

For each prime p, we have an exact sequence

$$0 \to (\mathcal{A}_{\mathbb{F}_p})^0 \to \mathcal{A}_{\mathbb{F}_p} \to \Phi_{A,p} \to 0,$$

where $(\mathcal{A}_{\mathbb{F}_p})^0$ is the connected compoent of the group scheme $\mathcal{A}_{\mathbb{F}_p}$, and $\Phi_{A,p}$ is a finite flat group scheme over $\mathbb{F}_p$ called the component group of A at p. The Tamagawa numbers c_p are then:

$$c_p = \#\Phi_{A,p}(\mathbb{F}_p),$$

and our goal is to compute them.

For each prime p with $p^2 \mid N$, the best we can do at present is use some *a priori* bounds on c_P. In particular, Lenstra and Oort [[ref]] prove that if a prime $\ell \mid c_p$, then either or $\ell = p$ or $\ell \leq 2d+1$. I do not know anything else about compute c_p in this case.

When $p \,||\, N$, I designed an algorithm involving quaternion algebras to compute the exact order of order of the component group $\Phi_{A,p}$ of the special fiber at p of the Néron model of A. Tate showed that if E is an elliptic curve with split multiplicative reduction at p, then there is an element $q \in \mathbb{Q}_p^*$ such that

$$Q_p^*/q^{\mathbb{Z}} \cong E(\mathbb{Q}_p).$$

In fact, something similar generalizes to any abelian variety A over $\mathbb{Q}$ that has *purely toric* reduction at p, i.e., such that the identity component $(\mathcal{A}_{\mathbb{F}_p})^0$ is isomorphic over $\overline{\mathbb{F}}_p$ to a product of copies of the multiplicative group $\mathbb{G}_m$. Any such A can then be viewed as a higher-dimensional analogue of a Tate curve, i.e., A has a *p-adic rigid analytic uniformization* over $\mathbb{Q}_p$:

$$0 \to X \to T \to A \to 0.$$

Here X is a free abelian group of rank $\dim(A)$, e.g., like $q^{\mathbb{Z}}$, and T is a torus (twist of product of copies of the multiplicative group $\mathbb{G}_m$). In the case when $A = A_f$ is a modular abelian variety, X and T can be described into terms of right ideal classes in an order of level N/p in the quaternion algebra ramified at p and ∞.

Let X_A be the free abelian group for A and $X_{A^\vee}$ the group for $A^\vee$. Then Grothendieck constructed a monodromy pairing

$$X_A \times X_{A^\vee} \to \mathbb{Z},$$

and one has an exact sequence

$$0 \to X_A \to \mathrm{Hom}(X_{A^\vee}, \mathbb{Z}) \to \Phi_{A,p} \to 0.$$

It is an open problem to compute the above exact sequence explicitly, but using a homogeneity trick one can at least compute enough to deduce $\#\Phi_{A,p}$. I will describe this homogeneity trick in more detail later [[ref]]. The main idea is that the association $A \mapsto X_A$ is functorial, so the Hecke operators act on X_A and on X_J, and there is a natural map $X_{A^\vee} \hookrightarrow X_J$. We *can* compute X_J explicitly, and use the Hecke operators to compute the saturated submodule $X_J[I_f]$, where $I_f = \mathrm{Ann}_{\mathbb{T}}(f)$. We have

$$X_{A^\vee} \subset X_J[I_f].$$

Though nobody knows how to compute $X_{A^\vee}$ yet, we do know how to compute $X_J[I_f]$. This uses that there is a very nice description of X_J as the group of degree zero divisors on the supersingular points of $X_0(N)_{\mathbb{F}_p}$, i.e., the pairs (E, C), where E is a supersingular elliptic curve over $\overline{\mathbb{F}}_p$ and C is a cyclic subgroup of E of order N/p.

The modular degree of A is the square root $m_A = \sqrt{\deg(A^\vee \to A)}$ of the degree of the natural map from $A^\vee$ to A induced by viewing A as a quotient of the Jacobian J.

Theorem 1.8. *Let $\alpha : X_J \to \mathrm{Hom}(X_J[I_f], \mathbb{Z})$ be the map induced by the monodromy pairing $X_J \times X_J \to \mathbb{Z}$. Then*

$$\#\Phi_{A,p} = \#\operatorname{coker}(\alpha) \cdot \frac{m_A}{\#(\alpha(X_J)/\alpha(X_J[I_f]))}$$

Remark 1.9. The image of the natural map $\Phi_{J,p} \to \Phi_{A,p}$ is of order $\#\operatorname{coker}(\alpha)$. The number $\frac{m_A}{\#(\alpha(X_J)/\alpha(X_J[I_f]))}$ is thus an integer, which measures congruences between the newform f and old forms in $S_2(\Gamma_0(N/p))$. This is because the modular degree measures all congruences, and $\#(\alpha(X_J)/\alpha(X_J[I_f]))$ measures congruences with forms of level divisible by p, as $X_J \otimes \mathbb{C}$ is isomorphic to the subspace of p-newforms in $S_2(\Gamma_0(N))$.

One can show $\mathrm{Gal}(\overline{\mathbb{F}}_p/\mathbb{F}_p)$ acts as either $+1$ or -1 on $\Phi_{A,p}(\overline{\mathbb{F}}_p)$. Then c_p is either $\#\Phi_{A,p}$ or $\#(\Phi_{A,p}[-1])$, depending on the eigenvalue of the Atkin-Lehner involution W_P acting on f. It is an open problem to obtain c_p exactly in general; it seems one might have to find an algorithm to compute the group structure of $\Phi_{A,p}$.

1.2.5 Open problems

In summary, solution to the following four open problems would result in an algorithm to compute the conjectural order of Ш(A). I have ordered the problems according to some vague sense of how hard they might be to solve.

1. Compute $\#A(\mathbb{Q})_{\mathrm{tor}} \cdot \#A^\vee(\mathbb{Q})_{\mathrm{tor}}$ exactly even when the above algorithm fails. For this, it would be helpful to understand precisely when the above algorithm fails.

2. Find an algorithm to compute the group structure of $\Phi_{A,p}$, hence be able to compute c_p.

3. Give an algorithm to compute c_A (or better, prove Manin's conjecture that $c_A = 1$ for all A).

4. Find an algorithm to compute c_p when $p^2 \mid N$.

2 Computing with Modular Symbols

2.1 Modular Symbols

Birch introduced modular symbols in the 1960s in order to study special values of L-functions when formulating the Birch and Swinnerton-Dyer conjecture. Modular symbols have grown to become perhaps the most important basic tool in computing with modular forms and modular abelian varieties. They are an explicit presentation for the homology group $\mathbb{H}_1(X_0(N), \mathbb{Z}, \{\text{cusps}\})$ of $X_0(N)$ relative to the cusps, which cares much deep arithmetic meaning. The best references for modular symbols are Cremona's book [[ref]], Merel's article in SLNM 1585 [[ref]], my Ph.D. thesis [[ref]], and my book [[ref]]. Lang's book [[ref]] is also interesting, though it doesn't view modular symbols as a computational tool.

[[The following is adapted from my book on modular forms.]]

Let $\boldsymbol{\mathcal{M}}_2$ be the *free abelian* group with basis the set of symbols $\{\alpha, \beta\}$ with

$$\alpha, \beta \in \mathbb{P}^1(\mathbb{Q}) = \mathbb{Q} \cup \{\infty\}$$

modulo the 3-term relations

$$\{\alpha, \beta\} + \{\beta, \gamma\} + \{\gamma, \alpha\} = 0$$

and modulo any torsion. Since $\boldsymbol{\mathcal{M}}_2$ is torsion-free, we use the above relation first with $\alpha = \beta = \gamma$ then with $\gamma = \alpha$ to deduce that

$$\{\alpha, \alpha\} = 0 \qquad \text{and} \qquad \{\alpha, \beta\} = -\{\beta, \alpha\}.$$

Remark 2.1 (Warning)**.** The symbols $\{\alpha, \beta\}$ satisfy the relations $\{\alpha, \beta\} = -\{\beta, \alpha\}$, so order matters. The notation $\{\alpha, \beta\}$ looks like the set containing two elements, which strongly (and incorrectly) suggests that the order does not matter. This is the standard notation in the literature.

We "think of" this modular symbol as the homology class, relative to the cusps, of a path from α to β in $\mathfrak{h}^*$.

Define a *left action of* $\mathrm{GL}_2(\mathbb{Q})$ on $\boldsymbol{\mathcal{M}}_2$ by letting $g \in \mathrm{GL}_2(\mathbb{Q})$ act by

$$g\{\alpha, \beta\} = \{g(\alpha), g(\beta)\},$$

and g acts on α and β via the corresponding linear fractional transformation. The space $\boldsymbol{\mathcal{M}}_2(\Gamma_0(N))$ of *modular symbols for* $\Gamma_0(N)$ is the quotient of $\boldsymbol{\mathcal{M}}_2$ by the submodule generated by the infinitely many elements of the form $x - g(x)$, for x in $\boldsymbol{\mathcal{M}}_2$ and g in $\Gamma_0(N)$, and modulo any torsion. A modular symbol for $\Gamma_0(N)$ is an element of this space. We frequently denote the equivalence class of a modular symbol by giving a representative element.

Example 2.2. Some modular symbols are 0 no matter what the level N is! For example, since $\gamma = \left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right) \in \Gamma_0(N)$, we have an equality of modular symbols for $\Gamma_0(N)$:

$$\{\infty, 0\} = \{\gamma(\infty), \gamma(0)\} = \{\infty, 1\},$$

so we have the following in the space of modular symbols for $\Gamma_0(N)$:

$$0 = \{\infty, 1\} - \{\infty, 0\} = \{\infty, 1\} + \{0, \infty\} = \{0, \infty\} + \{\infty, 1\} = \{0, 1\}.$$

Let $C_0(N) = \Gamma_0(N)\backslash\mathbb{P}^1(\mathbb{Q})$ be the set of cusps for $\Gamma_0(N)$, which is a finite set. There is a natural homomorphism

$$\varphi : \boldsymbol{\mathcal{M}}_2(\Gamma_0(N)) \to \mathbb{H}_1(X_0(N), C_0(N), \mathbb{Z}) \tag{2.1}$$

that sends a formal linear combination of (geodesic) paths in the upper half plane to their image as homology classes of paths on $X_0(N)$ with endpoints in the set of cusps. In [**?**] Manin proved that (2.1) is an *isomorphism* (this is a fairly involved topological argument).

Example 2.3. We illustrate modular symbols in the case when $N = 11$. Using Sage, which implements the algorithm that we describe below over $\mathbb{Q}$, we find that $\boldsymbol{\mathcal{M}}_2(\Gamma_0(11); \mathbb{Q})$ has basis $\{\infty, 0\}$, $\{-1/8, 0\}$, $\{-1/9, 0\}$. A basis for the integral homology $\mathbb{H}_1(X_0(11), \mathbb{Z})$ is the subgroup generated by $\{-1/8, 0\}$ and $\{-1/9, 0\}$.

```
sage: set_modsym_print_mode ('modular')
sage: M = ModularSymbols(11, 2)
sage: M.basis()
({Infinity,0}, {-1/8,0}, {-1/9,0})
sage: S = M.cuspidal_submodule()
sage: S.integral_basis()     # basis over ZZ.
({-1/8,0}, {-1/9,0})
sage: set_modsym_print_mode ('manin')    # set it back
```

2.2 Computing with Modular Symbols

[[The following is adapted from my book on modular forms.]]

2.2.1 Manin's Trick

In this section, we describe a trick of Manin that we will use to prove that spaces of modular symbols are computable.

The group $\Gamma_0(N)$ has finite index in $\mathrm{SL}_2(\mathbb{Z})$. Fix right coset representatives $r_0, r_1, \ldots, r_m$ for $\Gamma_0(N)$ in $\mathrm{SL}_2(\mathbb{Z})$, so that

$$\mathrm{SL}_2(\mathbb{Z}) = \Gamma_0(N)r_0 \cup \Gamma_0(N)r_1 \cup \cdots \cup \Gamma_0(N)r_m,$$

where the union is disjoint. For example, when N is prime, a list of coset representatives is

$$\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}, \begin{pmatrix}1 & 0\\ 1 & 1\end{pmatrix}, \begin{pmatrix}1 & 0\\ 2 & 1\end{pmatrix}, \begin{pmatrix}1 & 0\\ 3 & 1\end{pmatrix}, \ldots, \begin{pmatrix}1 & 0\\ N-1 & 1\end{pmatrix}, \begin{pmatrix}0 & -1\\ 1 & 0\end{pmatrix}.$$

Let

$$\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z}) = \{(a:b) \,:\, a,b \in \mathbb{Z}/N\mathbb{Z},\ \gcd(a,b,N) = 1\}/\sim \tag{2.2}$$

where $(a:b) \sim (a':b')$ if there is $u \in (\mathbb{Z}/N\mathbb{Z})^*$ such that $a = ua', b = ub'$.

Proposition 2.4. *There is a bijection between $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ and the right cosets of $\Gamma_0(N)$ in $\mathrm{SL}_2(\mathbb{Z})$, which sends a coset representative $\left(\begin{smallmatrix}a & b\\ c & d\end{smallmatrix}\right)$ to the class of $(c:d)$ in $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$.*

We now describe an observation of Manin (see [**?**, §1.5]) that is crucial to making $\boldsymbol{\mathcal{M}}_2(\Gamma_0(N))$ computable. It allows us to write any modular symbol $\{\alpha, \beta\}$ as a $\mathbb{Z}$-linear combination of symbols of the form $r_i\{0, \infty\}$, where the $r_i \in \mathrm{SL}_2(\mathbb{Z})$ are coset representatives as above. In particular, the finitely many symbols $r_0\{0,\infty\}, \ldots, r_m\{0,\infty\}$ generate $\boldsymbol{\mathcal{M}}_2(\Gamma_0(N))$.

Proposition 2.5 (Manin)**.** *Let N be a positive integer and $r_0, \ldots, r_m$ a set of right coset representatives for $\Gamma_0(N)$ in $\mathrm{SL}_2(\mathbb{Z})$. Every $\{\alpha, \beta\} \in \boldsymbol{\mathcal{M}}_2(\Gamma_0(N))$ is a $\mathbb{Z}$-linear combination of $r_0\{0,\infty\}, \ldots, r_m\{0,\infty\}$.*

We give two proofs of the proposition. The first is useful for computation (see [**?**, §2.1.6]); the second (see [**?**, §2]) is easier to understand conceptually since it does not require any knowledge of continued fractions.

Continued Fractions Proof of Proposition 2.5. Since

$$\{\alpha, \beta\} = \{0, \beta\} - \{0, \alpha\},$$

it suffices to consider modular symbols of the form $\{0, b/a\}$, where the rational number b/a is in lowest terms. Expand b/a as a continued fraction and consider the successive convergents in lowest terms:

$$\frac{b_{-2}}{a_{-2}} = \frac{0}{1}, \quad \frac{b_{-1}}{a_{-1}} = \frac{1}{0}, \quad \frac{b_0}{a_0} = \frac{b_0}{1}, \ldots, \quad \frac{b_{n-1}}{a_{n-1}}, \quad \frac{b_n}{a_n} = \frac{b}{a}$$

where the first two are included formally. Then

$$b_k a_{k-1} - b_{k-1} a_k = (-1)^{k-1},$$

so that

$$g_k = \begin{pmatrix} b_k & (-1)^{k-1} b_{k-1} \\ a_k & (-1)^{k-1} a_{k-1} \end{pmatrix} \in \mathrm{SL}_2(\mathbb{Z}).$$

Hence

$$\left\{ \frac{b_{k-1}}{a_{k-1}}, \frac{b_k}{a_k} \right\} = g_k\{0, \infty\} = r_i\{0, \infty\},$$

for some i, is of the required special form. Since

$$\{0, b/a\} = \{0, \infty\} + \{\infty, b_0\} + \left\{ \frac{b_0}{1}, \frac{b_1}{a_1} \right\} + \cdots + \left\{ \frac{b_{n-1}}{a_{n-1}}, \frac{b_n}{a_n} \right\},$$

this completes the proof. □

Inductive Proof of Proposition 2.5. As in the first proof it suffices to prove the proposition for any symbol $\{0, b/a\}$, where b/a is in lowest terms. We will induct on $a \in \mathbb{Z}_{\geq 0}$. If $a = 0$, then the symbol is $\{0, \infty\}$, which corresponds to the identity coset, so assume that $a > 0$. Find $a' \in \mathbb{Z}$ such that

$$ba' \equiv 1 \pmod{a};$$

then $b' = (ba' - 1)/a \in \mathbb{Z}$ so the matrix

$$\delta = \begin{pmatrix} b & b' \\ a & a' \end{pmatrix}$$

is an element of $\mathrm{SL}_2(\mathbb{Z})$. Thus $\delta = \gamma \cdot r_j$ for some right coset representative r_j and $\gamma \in \Gamma_0(N)$. Then

$$\{0, b/a\} - \{0, b'/a'\} = \{b'/a', b/a\} = \begin{pmatrix} b & b' \\ a & a' \end{pmatrix} \cdot \{0, \infty\} = r_j\{0, \infty\},$$

as elements of $\mathcal{M}_2(\Gamma_0(N))$. By induction, $\{0, b'/a'\}$ is a linear combination of symbols of the form $r_k\{0, \infty\}$, which completes the proof. □

Example 2.6. Let $N = 11$, and consider the modular symbol $\{0, 4/7\}$. We have

$$\frac{4}{7} = 0 + \frac{1}{1 + \frac{1}{1 + \frac{1}{3}}},$$

so the partial convergents are

$$\frac{b_{-2}}{a_{-2}} = \frac{0}{1}, \quad \frac{b_{-1}}{a_{-1}} = \frac{1}{0}, \quad \frac{b_0}{a_0} = \frac{0}{1}, \quad \frac{b_1}{a_1} = \frac{1}{1}, \quad \frac{b_2}{a_2} = \frac{1}{2}, \quad \frac{b_3}{a_3} = \frac{4}{7}.$$

Thus, noting as in Example 2.2 that $\{0,1\} = 0$, we have

$$\begin{aligned}
\{0,4/7\} &= \{0,\infty\} + \{\infty,0\} + \{0,1\} + \{1,1/2\} + \{1/2,4/7\} \\
&= \begin{pmatrix} 1 & -1 \\ 2 & -1 \end{pmatrix} \{0,\infty\} + \begin{pmatrix} 4 & 1 \\ 7 & 2 \end{pmatrix} \{0,\infty\} \\
&= \begin{pmatrix} 1 & 0 \\ 9 & 1 \end{pmatrix} \{0,\infty\} + \begin{pmatrix} 1 & 0 \\ 9 & 1 \end{pmatrix} \{0,\infty\} \\
&= 2 \cdot \left[\begin{pmatrix} 1 & 0 \\ 9 & 1 \end{pmatrix} \{0,\infty\} \right].
\end{aligned}$$

We compute the convergents of $4/7$ in Sage as follows (note that 0 and ∞ are excluded):

```
sage: convergents(4/7)
[0, 1, 1/2, 4/7]
```

2.3 Manin Symbols

[[The following is adapted from my book on modular forms.]]

As above, fix coset representatives $r_0, \ldots, r_m$ for $\Gamma_0(N)$ in $\mathrm{SL}_2(\mathbb{Z})$. Consider formal symbols $[r_i]'$ for $i = 0, \ldots, m$. Let $[r_i]$ be the modular symbol $r_i\{0,\infty\} = \{r_i(0), r_i(\infty)\}$. We equip the symbols $[r_0]', \ldots, [r_m]'$ with a *right action of* $\mathrm{SL}_2(\mathbb{Z})$, which is given by $[r_i]'.g = [r_j]'$, where $\Gamma_0(N) r_j = \Gamma_0(N) r_i g$. We extend the notation by writing $[\gamma]' = [\Gamma_0(N)\gamma]' = [r_i]'$, where $\gamma \in \Gamma_0(N) r_i$. Then the right action of $\Gamma_0(N)$ is simply $[\gamma]'.g = [\gamma g]'$.

The group $\mathrm{SL}_2(\mathbb{Z})$ is generated [[ref]] by the two matrices $\sigma = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$ and $\tau = \left(\begin{smallmatrix} 1 & -1 \\ 1 & 0 \end{smallmatrix}\right)$. Note that $\sigma = S$ and $\tau = TS$, so $T = \tau\sigma \in \langle \sigma, \tau \rangle$.

The following theorem provides us with a finite presentation for the space $\mathcal{M}_2(\Gamma_0(N))$ of modular symbols.

Theorem 2.7 (Manin)**.** *Consider the quotient M of the free abelian group on Manin symbols $[r_0]', \ldots, [r_m]'$ by the subgroup generated by the elements (for all i):*

$$[r_i]' + [r_i]'\sigma \qquad \text{and} \qquad [r_i]' + [r_i]'\tau + [r_i]'\tau^2,$$

and modulo any torsion. Then there is an isomorphism

$$\Psi : M \xrightarrow{\sim} \boldsymbol{\mathcal{M}}_2(\Gamma_0(N))$$

given by $[r_i]' \mapsto [r_i] = r_i\{0,\infty\}$.

Proof. We will only prove that Ψ is surjective; the proof that Ψ is injective requires much more work and will be omitted from this book (see [**?**, §1.7] for a complete proof).

Proposition 2.5 implies that Ψ is surjective, assuming that Ψ is well defined. We next verify that Ψ is well defined, i.e., that the listed 2-term and 3-term

relations hold in the image. To see that the first relation holds, note that

$$\begin{aligned}[r_i] + [r_i]\sigma &= \{r_i(0), r_i(\infty)\} + \{r_i\sigma(0), r_i\sigma(\infty)\} \\ &= \{r_i(0), r_i(\infty)\} + \{r_i(\infty), r_i(0)\} \\ &= 0.\end{aligned}$$

For the second relation we have

$$\begin{aligned}[r_i] + [r_i]\tau + [r_i]\tau^2 &= \{r_i(0), r_i(\infty)\} + \{r_i\tau(0), r_i\tau(\infty)\} + \{r_i\tau^2(0), r_i\tau^2(\infty)\} \\ &= \{r_i(0), r_i(\infty)\} + \{r_i(\infty), r_i(1)\} + \{r_i(1), r_i(0)\} \\ &= 0.\end{aligned}$$

□

Example 2.8. By default Sage computes modular symbols spaces over $\mathbb{Q}$, i.e., $\boldsymbol{\mathcal{M}}_2(\Gamma_0(N); \mathbb{Q}) \cong \boldsymbol{\mathcal{M}}_2(\Gamma_0(N)) \otimes \mathbb{Q}$. Sage represents (weight 2) Manin symbols as pairs (c, d). Here c, d are integers that satisfy $0 \leq c, d < N$; they define a point $(c : d) \in \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$, hence a right coset of $\Gamma_0(N)$ in $\mathrm{SL}_2(\mathbb{Z})$ (see Proposition 2.4).

Create $\boldsymbol{\mathcal{M}}_2(\Gamma_0(N); \mathbb{Q})$ in Sage by typing `ModularSymbols(N, 2)`. We then use the Sage command `manin_generators` to enumerate a list of generators $[r_0], \ldots, [r_n]$ as in Theorem 2.7 for several spaces of modular symbols.

```
sage: M = ModularSymbols(2,2)
sage: M
Modular Symbols space of dimension 1 for Gamma_0(2)
of weight 2 with sign 0 over Rational Field
sage: M.manin_generators()
[(0,1), (1,0), (1,1)]

sage: M = ModularSymbols(3,2)
sage: M.manin_generators()
[(0,1), (1,0), (1,1), (1,2)]

sage: M = ModularSymbols(6,2)
sage: M.manin_generators()
[(0,1), (1,0), (1,1), (1,2), (1,3), (1,4), (1,5), (2,1),
 (2,3), (2,5), (3,1), (3,2)]
```

Given `x=(c,d)`, the command `x.lift_to_sl2z(N)` computes an element of $\mathrm{SL}_2(\mathbb{Z})$ whose lower two entries are congruent to (c, d) modulo N.

```
sage: M = ModularSymbols(2,2)
sage: [x.lift_to_sl2z(2) for x in M.manin_generators()]
[[1, 0, 0, 1], [0, -1, 1, 0], [0, -1, 1, 1]]
sage: M = ModularSymbols(6,2)
sage: x = M.manin_generators()[9]
sage: x
```

```
(2,5)
sage: x.lift_to_sl2z(6)
[1, 2, 2, 5]
```

The `manin_basis` command returns a list of indices into the Manin generator list such that the corresponding symbols form a basis for the quotient of the $\mathbb{Q}$-vector space spanned by Manin symbols modulo the 2-term and 3-term relations of Theorem 2.7.

```
sage: M = ModularSymbols(2,2)
sage: M.manin_basis()
[1]
sage: [M.manin_generators()[i] for i in M.manin_basis()]
[(1,0)]
sage: M = ModularSymbols(6,2)
sage: M.manin_basis()
[1, 10, 11]
sage: [M.manin_generators()[i] for i in M.manin_basis()]
[(1,0), (3,1), (3,2)]
```

Thus, e.g., every element of $\boldsymbol{\mathcal{M}}_2(\Gamma_0(6))$ is a $\mathbb{Q}$-linear combination of the three symbols $[(1,0)]$, $[(3,1)]$, and $[(3,2)]$. We can write each of these as a modular symbol using the `modular_symbol_rep` function.

```
sage: M.basis()
((1,0), (3,1), (3,2))
sage: [x.modular_symbol_rep() for x in M.basis()]
[{Infinity,0}, {0,1/3}, {-1/2,-1/3}]
```

The `manin_gens_to_basis` function returns a matrix whose rows express each Manin symbol generator in terms of the subset of Manin symbols that forms a basis (as returned by `manin_basis`).

```
sage: M = ModularSymbols(2,2)
sage: M.manin_gens_to_basis()
[-1]
[ 1]
[ 0]
```

Since the basis is $(1,0)$, this means that in $\boldsymbol{\mathcal{M}}_2(\Gamma_0(2);\mathbb{Q})$, we have $[(0,1)] = -[(1,0)]$ and $[(1,1)] = 0$. (Since no denominators are involved, we have in fact computed a presentation of $\boldsymbol{\mathcal{M}}_2(\Gamma_0(2);\mathbb{Z})$.)

To convert a Manin symbol $x = (c,d)$ to an element of a modular symbols space M, use `M(x)`:

```
sage: M = ModularSymbols(2,2)
sage: x = (1,0); M(x)
(1,0)
```

Next consider $\boldsymbol{\mathcal{M}}_2(\Gamma_0(6);\mathbb{Q})$:

```
sage: M = ModularSymbols(6,2)
sage: M.manin_gens_to_basis()
[-1  0  0]
[ 1  0  0]
[ 0  0  0]
[ 0 -1  1]
[ 0 -1  0]
[ 0 -1  1]
[ 0  0  0]
[ 0  1 -1]
[ 0  0 -1]
[ 0  1 -1]
[ 0  1  0]
[ 0  0  1]
```

Recall that our choice of basis for $\boldsymbol{\mathcal{M}}_2(\Gamma_0(6);\mathbb{Q})$ is $[(1,0)], [(3,1)], [(3,2)]$. Thus, e.g., the first row of this matrix says that $[(0,1)] = -[(1,0)]$, and the fourth row asserts that $[(1,2)] = -[(3,1)] + [(3,2)]$.

```
sage: M = ModularSymbols(6,2)
sage: M((0,1))
-(1,0)
sage: M((1,2))
-(3,1) + (3,2)
```

2.4 Hecke Operators

[[The following is adapted from my book on modular forms.]]

2.4.1 Hecke Operators on Modular Symbols

When p is a prime not dividing N, define

$$T_p(\{\alpha,\beta\}) = \begin{pmatrix} p & 0 \\ 0 & 1 \end{pmatrix} \{\alpha,\beta\} + \sum_{r \bmod p} \begin{pmatrix} 1 & r \\ 0 & p \end{pmatrix} \{\alpha,\beta\}.$$

The Hecke operators are compatible with the integration pairing $\langle\ ,\ \rangle$, in the sense that $\langle fT_p, x\rangle = \langle f, T_p x\rangle$. When $p \mid N$, the definition is the same, except that the matrix $\left(\begin{smallmatrix} p & 0 \\ 0 & 1 \end{smallmatrix}\right)$ is not included in the sum. There is a similar definition of T_n for n composite.

Example 2.9. For example, when $N = 11$, we have

$$\begin{aligned} T_2\{0,1/5\} &= \{0,2/5\} + \{0,1/10\} + \{1/2,3/5\} \\ &= -2\{0,1/5\}. \end{aligned}$$

2.5 Hecke Operators on Manin Symbols

In [**?**], L. Merel gives a description of the action of T_p directly on Manin symbols $[r_i]$. For example, when $p = 2$ and N is odd, we have

$$T_2([r_i]) = [r_i]\begin{pmatrix}1 & 0\\ 0 & 2\end{pmatrix} + [r_i]\begin{pmatrix}2 & 0\\ 0 & 1\end{pmatrix} + [r_i]\begin{pmatrix}2 & 1\\ 0 & 1\end{pmatrix} + [r_i]\begin{pmatrix}1 & 0\\ 1 & 2\end{pmatrix}. \tag{2.3}$$

For any prime, let C_p be the set of matrices constructed using the following algorithm (see [**?**, §2.4]):

Algorithm 2.10 (Cremona's Heilbronn Matrices)**.** Given an *odd* prime p, this algorithm outputs a list of 2×2 matrices of determinant p that can be used to compute the Hecke operator T_p.

1. Output $\begin{pmatrix}1 & 0\\ 0 & p\end{pmatrix}$.
2. For $r = \left\lceil -\frac{p}{2} \right\rceil, \ldots, \left\lfloor \frac{p}{2} \right\rfloor$:
   (a) Let $x_1 = p$, $x_2 = -r$, $y_1 = 0$, $y_2 = 1$, $a = -p$, $b = r$.
   (b) Output $\begin{pmatrix}x_1 & x_2\\ y_1 & y_2\end{pmatrix}$.
   (c) As long as $b \neq 0$, do the following:
      i. Let q be the integer closest to a/b (if a/b is a half integer, round away from 0).
      ii. Let $c = a - bq$, $a = -b$, $b = c$.
      iii. Set $x_3 = qx_2 - x_1$, $x_1 = x_2$, $x_2 = x_3$, and $y_3 = qy_2 - y_1$, $y_1 = y_2$, $y_2 = y_3$.
      iv. Output $\begin{pmatrix}x_1 & x_2\\ y_1 & y_2\end{pmatrix}$.

Proposition 2.11 (Cremona, Merel)**.** *Let C_p be as above. Then for $p \nmid N$ and $[x] \in \mathcal{M}_2(\Gamma_0(N))$ a Manin symbol, we have*

$$T_p([x]) = \sum_{g \in C_p} [xg].$$

Proof. See Proposition 2.4.1 of [**?**]. □

There are other lists of matrices, due to Merel, that work even when $p \mid N$.

The command `HeilbronnCremonaList(p)`, for p prime, outputs the list of matrices from Algorithm 2.10.

```
sage: HeilbronnCremonaList(2)
[[1, 0, 0, 2], [2, 0, 0, 1], [2, 1, 0, 1], [1, 0, 1, 2]]
sage: HeilbronnCremonaList(3)
[[1, 0, 0, 3], [3, 1, 0, 1], [1, 0, 1, 3], [3, 0, 0, 1],
```

```
 [3, -1, 0, 1], [-1, 0, 1, -3]]
sage: HeilbronnCremonaList(5)
[[1, 0, 0, 5], [5, 2, 0, 1], [2, 1, 1, 3], [1, 0, 3, 5],
 [5, 1, 0, 1], [1, 0, 1, 5], [5, 0, 0, 1], [5, -1, 0, 1],
 [-1, 0, 1, -5], [5, -2, 0, 1], [-2, 1, 1, -3],
 [1, 0, -3, 5]]
sage: len(HeilbronnCremonaList(37))
128
sage: len(HeilbronnCremonaList(389))
1892
sage: len(HeilbronnCremonaList(2003))
11662
```

Example 2.12. We compute the matrix of T_2 on $\boldsymbol{\mathcal{M}}_2(\Gamma_0(2))$:

```
sage: M = ModularSymbols(2,2)
sage: M.T(2).matrix()
[1]
```

Example 2.13. We compute some Hecke operators on $\boldsymbol{\mathcal{M}}_2(\Gamma_0(6))$:

```
sage: M = ModularSymbols(6, 2)
sage: M.T(2).matrix()
[ 2  1 -1]
[-1  0  1]
[-1 -1  2]
sage: M.T(3).matrix()
[3 2 0]
[0 1 0]
[2 2 1]
sage: M.T(3).fcp()  # factored characteristic polynomial
(x - 3) * (x - 1)^2
```

For $p \geq 5$ we have $T_p = p + 1$, since $M_2(\Gamma_0(6))$ is spanned by generalized Eisenstein series (see Chapter **??**).

Example 2.14. We compute the Hecke operators on $\boldsymbol{\mathcal{M}}_2(\Gamma_0(39))$:

```
sage: M = ModularSymbols(39, 2)
sage: T2 = M.T(2)
sage: T2.matrix()
[ 3  0 -1  0  0  1  1 -1  0]
[ 0  0  2  0 -1  1  0  1 -1]
[ 0  1  0 -1  1  1  0  1 -1]
[ 0  0  1  0  0  1  0  1 -1]
[ 0 -1  2  0  0  1  0  1 -1]
[ 0  0  1  1  0  1  1 -1  0]
[ 0  0  0 -1  0  1  1  2  0]
```

```
[ 0  0  0  1  0  0  2  0  1]
[ 0  0 -1  0  0  0  1  0  2]
sage: T2.fcp()     # factored characteristic polynomial
(x - 3)^3 * (x - 1)^2 * (x^2 + 2*x - 1)^2
```

The Hecke operators commute, so their eigenspace structures are related.

```
sage: T2 = M.T(2).matrix()
sage: T5 = M.T(5).matrix()
sage: T2*T5 - T5*T2 == 0
True
sage: T5.charpoly().factor()
(x^2 - 8)^2 * (x - 6)^3 * (x - 2)^2
```

The decomposition of T_2 is a list of the kernels of $(f^e)(T_2)$, where f runs through the irreducible factors of the characteristic polynomial of T_2 and f^e exactly divides this characteristic polynomial. Using Sage, we find them:

```
sage: M = ModularSymbols(39, 2)
sage: M.T(2).decomposition()
[
Modular Symbols subspace of dimension 3 of Modular
Symbols space of dimension 9 for Gamma_0(39) of weight
2 with sign 0 over Rational Field,
Modular Symbols subspace of dimension 2 of Modular
Symbols space of dimension 9 for Gamma_0(39) of weight
2 with sign 0 over Rational Field,
Modular Symbols subspace of dimension 4 of Modular
Symbols space of dimension 9 for Gamma_0(39) of weight
2 with sign 0 over Rational Field
]
```

2.6 A Complete toy Implementation

In this section we give a complete high-level toy SAge implementation of the above algorithm for computing modular symbols and Hecke operators on them when the level is prime.

```
def reduce(v):
    """Return index into the list (0,1), (1,0), (1,1), ...
        of equivalent symbol."""
    if v[0]: return 1+int(v[1]/v[0])
    return 0

def manin_symbols(p):
    """Return list of all Manin symbols (c,d)."""
    V = GF(p)^2
```

```
    return [V([0,1])] + [V([1,a]) for a in GF(p)]

def relation_matrix(p):
    """Returns sparse matrix of relations between Manin symbols
       of level Gamma0(p)."""
    syms = manin_symbols(p)
    sigma = matrix(GF(p), 2, [0,-1,1,0])
    tau = matrix(GF(p), 2, [1,-1,1,0]); tau2 = tau*tau
    A = matrix(QQ, 2*len(syms), len(syms), sparse=True)
    i = 0
    for v in syms:
        j = reduce(v)
        A[i,j] = 1; A[i,reduce(v*sigma)] += 1;    i += 1
        A[i,j] = 1; A[i,reduce(v*tau)] += 1; A[i,reduce(v*tau2)] += 1;
        i += 1
    return A

class modular_symbols:
    """Space of toy modular symbols of prime level p."""
    def __init__(self, p):
        assert is_prime(p)
        self.syms = manin_symbols(p)
        rels = relation_matrix(p).row_module()
        self.Q = rels.ambient_module() / rels    # quotient vector space
        self.p = p; self.dim = self.Q.dimension()

    def __repr__(self):
        return "Toy modular symbols of level %s and dimension %s"%(
              self.p, self.dim)

    def free_gens(self):
        """Return indexes of freely generating Manin symbols."""
        return [self.Q.lift(self.Q.gen(i)).nonzero_positions()[0]
                     for i in range(self.dim)]

    def T(self, q, i):
        """Return T_q(v) with q!=p prime where v is the ith
           standard Manin symbol."""
        v = self.syms[i]; Q = self.Q; V = Q.V()
        return sum([Q( V.gen(reduce(v*h)) ) for h in
                      heilbronn_matrices(q, self.p)])

    def hecke_matrix(self, q):
        """Return matrix of q-th Hecke operator."""
        assert q != self.p
        return matrix([self.T(q,i) for i in self.free_gens()])
```

```
def heilbronn_matrices(p, m):
    """Return matrices of Heilbronn matrices of determinant p modulo m."""
    p = Integer(p); M = MatrixSpace(GF(m),2)
    if p == 2: return [M(z) for z in ([
                 [1,0,0,2],[2,0,0,1],[2,1,0,1],[1,0,1,2]])]
    v = [M([1,0,0,p])]
    for r in [ceil(-p/2)..floor(p/2)]:
        x1 = p; x2 = -r; y1=0; y2=1; a=-p; b=r
        v.append(M([x1,x2,y1,y2]))
        while b!=0:
            q = (a/b).round(mode='away')
            c = a-b*q; a=-b; b=c
            x3 = q*x2-x1; x1=x2; x2=x3; y3=q*y2-y1; y1=y2; y2=y3
            v.append(M([x1,x2,y1,y2]))
    return v
```

2.7 Cuspidal Modular Symbols and the Boundary Map

Manin identified the subspace of $\boldsymbol{\mathcal{M}}_2(\Gamma_0(N))$ that is sent isomorphically onto $\mathbb{H}_1(X_0(N), \mathbb{Z})$. Let $\boldsymbol{\mathcal{B}}_2(\Gamma_0(N))$ denote the free abelian group whose basis is the set $C_0(N)$ of cusps for $\Gamma_0(N)$. The *boundary map*

$$\delta : \boldsymbol{\mathcal{M}}_2(\Gamma_0(N)) \to \boldsymbol{\mathcal{B}}_2(\Gamma_0(N))$$

sends $\{\alpha, \beta\}$ to $\{\beta\} - \{\alpha\}$, where $\{\beta\}$ denotes the basis element of $\boldsymbol{\mathcal{B}}_2(\Gamma_0(N))$ corresponding to $\beta \in \mathbb{P}^1(\mathbb{Q})$. The kernel $\boldsymbol{\mathcal{S}}_2(\Gamma_0(N))$ of δ is the subspace of *cuspidal modular symbols*. Thus an element of $\boldsymbol{\mathcal{S}}_2(\Gamma_0(N))$ can be thought of as a linear combination of paths in $\mathfrak{h}^*$ whose endpoints are cusps and whose images in $X_0(N)$ are homologous to a $\mathbb{Z}$-linear combination of closed paths.

Theorem 2.15 (Manin)**.** *The map φ above induces a canonical isomorphism*

$$\boldsymbol{\mathcal{S}}_2(\Gamma_0(N)) \cong \mathbb{H}_1(X_0(N), \mathbb{Z}).$$

Proof. This is [**?**, Thm. 1.9]. □

For any (commutative) ring R let

$$\boldsymbol{\mathcal{M}}_2(\Gamma_0(N), R) = \boldsymbol{\mathcal{M}}_2(\Gamma_0(N)) \otimes_{\mathbb{Z}} R$$

and

$$\boldsymbol{\mathcal{S}}_2(\Gamma_0(N), R) = \boldsymbol{\mathcal{S}}_2(\Gamma_0(N)) \otimes_{\mathbb{Z}} R.$$

Proposition 2.16. *We have*

$$\dim_{\mathbb{C}} \boldsymbol{\mathcal{S}}_2(\Gamma_0(N), \mathbb{C}) = 2 \dim_{\mathbb{C}} S_2(\Gamma_0(N)).$$

Proof. We have

$$\dim_{\mathbb{C}} \boldsymbol{\mathcal{S}}_2(\Gamma_0(N), \mathbb{C}) = \operatorname{rank}_{\mathbb{Z}} \boldsymbol{\mathcal{S}}_2(\Gamma_0(N)) = \operatorname{rank}_{\mathbb{Z}} \mathbb{H}_1(X_0(N), \mathbb{Z}) = 2g.$$

□

To compute the boundary map on $[\gamma]$, note that $[\gamma] = \{\gamma(0), \gamma(\infty)\}$, so if $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$, then

$$\delta([\gamma]) = \{\gamma(\infty)\} - \{\gamma(0)\} = \{a/c\} - \{b/d\}.$$

Computing this boundary map would appear to first require an algorithm to compute the set $C(\Gamma_0(N)) = \Gamma_0(N)\backslash\mathbb{P}^1(\mathbb{Q})$ of cusps for $\Gamma_0(N)$. (Given such an algorithm is not difficult.) However, there is a trick that computes the set of cusps in the course of running the algorithm. First, give an algorithm for deciding whether or not two elements of $\mathbb{P}^1(\mathbb{Q})$ are equivalent modulo the action of $\Gamma_0(N)$. Then simply construct $C(\Gamma_0(N))$ in the course of computing the boundary map, i.e., keep a list of cusps found so far, and whenever a new cusp class is discovered, add it to the list. The following proposition, which is proved in [**?**, Prop. 2.2.3], explains how to determine whether two cusps are equivalent.

Proposition 2.17 (Cremona)**.** *Let (c_i, d_i), $i = 1, 2$, be pairs of integers with $\gcd(c_i, d_i) = 1$ and possibly $d_i = 0$. There is $g \in \Gamma_0(N)$ such that $g(c_1/d_1) = c_2/d_2$ in $\mathbb{P}^1(\mathbb{Q})$ if and only if*

$$s_1 d_2 \equiv s_2 d_1 \pmod{\gcd(d_1 d_2, N)}$$

where s_j satisfies $c_j s_j \equiv 1 \pmod{d_j}$.

In Sage the command `boundary_map()` computes the boundary map from $\boldsymbol{\mathcal{M}}_2(\Gamma_0(N))$ to $\boldsymbol{\mathcal{B}}_2(\Gamma_0(N))$, and the `cuspidal_submodule()` command computes its kernel. For example, for level 2 the boundary map is given by the matrix $[1 \ \ -1]$, and its kernel is the 0 space:

```
sage: M = ModularSymbols(2, 2)
sage: M.boundary_map()
Hecke module morphism boundary map defined by the matrix
[ 1 -1]
Domain: Modular Symbols space of dimension 1 for
Gamma_0(2) of weight ...
Codomain: Space of Boundary Modular Symbols for
Congruence Subgroup Gamma0(2) ...
sage: M.cuspidal_submodule()
Modular Symbols subspace of dimension 0 of Modular
Symbols space of dimension 1 for Gamma_0(2) of weight
2 with sign 0 over Rational Field
```

The smallest level for which the boundary map has nontrivial kernel, i.e., for which $\boldsymbol{\mathcal{S}}_2(\Gamma_0(N)) \neq 0$, is $N = 11$.

```
sage: M = ModularSymbols(11, 2)
sage: M.boundary_map().matrix()
[ 1 -1]
[ 0  0]
[ 0  0]
sage: M.cuspidal_submodule()
Modular Symbols subspace of dimension 2 of Modular
Symbols space of dimension 3 for Gamma_0(11) of weight
2 with sign 0 over Rational Field
sage: S = M.cuspidal_submodule(); S
Modular Symbols subspace of dimension 2 of Modular
Symbols space of dimension 3 for Gamma_0(11) of weight
2 with sign 0 over Rational Field
sage: S.basis()
((1,8), (1,9))
```

The following illustrates that the Hecke operators preserve $\boldsymbol{\mathcal{S}}_2(\Gamma_0(N))$:

```
sage: S.T(2).matrix()
[-2  0]
[ 0 -2]
sage: S.T(3).matrix()
[-1  0]
[ 0 -1]
sage: S.T(5).matrix()
[1 0]
[0 1]
```

A nontrivial fact is that for p prime the eigenvalue of each of these matrices is $p + 1 - \#E(\mathbb{F}_p)$, where E is the elliptic curve $X_0(11)$ defined by the (affine) equation $y^2 + y = x^3 - x^2 - 10x - 20$. For example, we have

```
sage: E = EllipticCurve([0,-1,1,-10,-20])
sage: 2 + 1 - E.Np(2)
-2
sage: 3 + 1 - E.Np(3)
-1
sage: 5 + 1 - E.Np(5)
1
sage: 7 + 1 - E.Np(7)
-2
```

The same numbers appear as the eigenvalues of Hecke operators:

```
sage: [S.T(p).matrix()[0,0] for p in [2,3,5,7]]
[-2, -1, 1, -2]
```

In fact, something similar happens for every elliptic curve over $\mathbb{Q}$. The book [**?**] (especially Chapter 8) is about this striking numerical relationship between the number of points on elliptic curves modulo p and coefficients of modular forms.

2.8 Newforms: Systems of Eigenvalues

In this section we describe an algorithm for computing the system of Hecke eigenvalues associated to a simple subspace of a space of modular symbols. This algorithm is better than doing linear algebra directly over the number field generated by the eigenvalues. It only involves linear algebra over the base field and also yields a compact representation for the answer, which is better than writing the eigenvalues in terms of a power basis for a number field. In order to use this algorithm, it is necessary to decompose the space of cuspidal modular symbols as a direct sum of simples.

Fix N and a Dirichlet character ε of modulus N, and let

$$V = \boldsymbol{\mathcal{M}}_2(N, \varepsilon)^+$$

be the $+1$ quotient of modular symbols, i.e, the submodule fixed by the $*$ involution given by

$$*\{\alpha, \beta\} = \{-\alpha, -\beta\}.$$

Algorithm 2.18 (System of Eigenvalues)**.** Given a $\mathbb{T}$-simple subspace $W \subset V$ of modular symbols, this algorithm outputs maps ψ and e, where $\psi : \mathbb{T}_K \to W$ is a K-linear map and $e : W \cong L$ is an isomorphism of W with a number field L, such that $a_n = e(\psi(T_n))$ is the eigenvalue of the nth Hecke operator acting on a fixed $\mathbb{T}$-eigenvector in $W \otimes \overline{\mathbb{Q}}$. (Thus $f = \sum_{n=1}^{\infty} e(\psi(T_n))q^n$ is a newform.)

1. [Compute Projection] Let $\varphi : V \to W'$ be any surjective linear map such that $\ker(\varphi)$ equals the kernel of the $\mathbb{T}$-invariant projection onto W. For example, compute φ by finding a simple submodule of $V^* = \mathrm{Hom}(V, K)$ that is isomorphic to W, e.g., by cutting out eigenspaces with in V^* with T replaced by the transpose of T.
2. [Choose v] Choose a nonzero element $v \in V$ such that $\pi(v) \neq 0$ and computation of $T_n(v)$ is "easy", e.g., choose v to be a Manin symbol.
3. [Map from Hecke Ring] Let ψ be the map $\mathbb{T} \to W'$, given by $\psi(t) = \pi(tv)$. Note that computation of ψ is relatively easy, because v was chosen so that tv is relatively easy to compute. In particular, if $t = T_p$, we do not need to compute the full matrix of T_p on V; instead we just compute $T_p(v)$.
4. [Find Generator] Find a random $T \in \mathbb{T}$ such that the iterates

   $$\psi(T^0), \quad \psi(T), \quad \psi(T^2), \quad \ldots, \quad \psi(T^{d-1})$$

   are a basis for W', where W has dimension d.
5. [Characteristic Polynomial] Compute the characteristic polynomial f of $T|_W$, and let $L = K[x]/(f)$. Because of how we chose T in step (4), the minimal and characteristic polynomials of $T|_W$ are equal, and both are irreducible, so L is an extension of K of degree $d = \dim(W)$.
6. [Field Structure] In this step we endow W' with a field structure. Let $e : W' \to L$ be the unique K-linear isomorphism such that

   $$e(\psi(T^i)) \equiv x^i \pmod{f}$$

for $i = 0, 1, 2, ..., \deg(f) - 1$. The map e is uniquely determined since the $\psi(T^i)$ are a basis for W'. To compute e, we compute the change of basis matrix from the standard basis for W' to the basis $\{\psi(T^i)\}$. This change of basis matrix is the inverse of the matrix whose rows are the $\psi(T^i)$ for $i = 0, ..., \deg(f) - 1$.

7. [Hecke Eigenvalues] Finally for each integer $n \geq 1$, we have

$$a_n = e(\psi(T_n)) = e(\pi(T_n(v))),$$

where a_n is the eigenvalue of T_n. Output the maps ψ and e and terminate.

One reason we separate ψ and e is that when $\dim(W)$ is large, the values $\psi(T_n)$ take less space to store and are easier to compute, whereas each one of the values $e(\psi(n))$ is huge.[1] The function e typically involves large numbers if $\dim(W)$ is large, since e is obtained from the iterates of a single vector. For many applications, e.g., databases, it is better to store a matrix that defines e and the images under ψ of many T_n.

Example 2.19. The space $S_2(\Gamma_0(23))$ of cusp forms has dimension 2 and is spanned by two $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$-conjugate newforms, one of which is

$$f = q + aq^2 + (-2a - 1)q^3 + (-a - 1)q^4 + 2aq^5 + \cdots,$$

where $a = (-1 + \sqrt{5})/2$. We will use Algorithm 2.18 to compute a few of these coefficients.

The space $\boldsymbol{\mathcal{M}}_2(\Gamma_0(23))^+$ of modular symbols has dimension 3. It has the following basis of Manin symbols:

$$[(0, 0)], \quad [(1, 0)], \quad [(0, 1)],$$

where we use square brackets to differentiate Manin symbols from vectors. The Hecke operator

$$T_2 = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 0 & 2 \\ -1 & 1/2 & -1 \end{pmatrix}$$

has characteristic polynomial $(x-3)(x^2+x-1)$. The kernel of T_2-3 corresponds to the span of the Eisenstein series of level 23 and weight 2, and the kernel V of $T_2^2 + T_2 - 1$ corresponds to $S_2(\Gamma_0(23))$. (We could also have computed V as the kernel of the boundary map $\boldsymbol{\mathcal{M}}_2(\Gamma_0(23))^+ \to \boldsymbol{\mathcal{B}}_2(\Gamma_0(23))^+$.) Each of the following steps corresponds to the step of Algorithm 2.18 with the same number.

1. [Compute Projection] We compute projection onto V (this will suffice to give us a map ϕ as in the algorithm). The matrix whose first two columns are the echelon basis for V and whose last column is the echelon basis for the Eisenstein subspace is

$$\begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & -2/11 \\ 0 & 1 & -3/11 \end{pmatrix}$$

[1] John Cremona initially suggested to me the idea of separating these two maps.

and

$$B^{-1} = \begin{pmatrix} 2/11 & 1 & 0 \\ 3/11 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix},$$

so projection onto V is given by the first two rows:

$$\pi = \begin{pmatrix} 2/11 & 1 & 0 \\ 3/11 & 0 & 1 \end{pmatrix}.$$

2. [Choose v] Let $v = (0,1,0)^t$. Notice that $\pi(v) = (1,0)^t \neq 0$, and $v = [(1,0)]$ is a sum of only one Manin symbol.

3. [Map from Hecke Ring] This step is purely conceptual, since no actual work needs to be done. We illustrate it by computing $\psi(T_1)$ and $\psi(T_2)$. We have
$$\psi(T_1) = \pi(v) = (1,0)^t$$
and
$$\psi(T_2) = \pi(T_2(v)) = \pi((0,0,1/2)^t) = (0,1/2)^t.$$

4. [Find Generator] We have
$$\psi(T_2^0) = \psi(T_1) = (1,0)^t,$$
which is clearly independent from $\psi(T_2) = (0,1/2)^t$. Thus we find that the image of the powers of $T = T_2$ generate V.

5. [Characteristic Polynomial] The matrix of $T_2|_V$ is $\left(\begin{smallmatrix} 0 & 2 \\ 1/2 & -1 \end{smallmatrix}\right)$, which has characteristic polynomial $f = x^2 + x - 1$. Of course, we already knew this because we computed V as the kernel of $T_2^2 + T_2 - 1$.

6. [Field Structure] We have
$$\psi(T_2^0) = \pi(v) = (1,0)^t \text{ and } \psi(T_2) = (0,1/2).$$
The matrix with rows the $\psi(T_2^i)$ is $\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1/2 \end{smallmatrix}\right)$, which has inverse $e = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 2 \end{smallmatrix}\right)$. The matrix e defines an isomorphism between V and the field
$$L = \mathbb{Q}[x]/(f) = \mathbb{Q}((-1+\sqrt{5})/2).$$
I.e., $e((1,0)) = 1$ and $e((0,1)) = 2x$, where $x = (-1+\sqrt{5})/2$.

7. [Hecke Eigenvalues] We have $a_n = e(\Psi(T_n))$. For example,

$$\begin{aligned}
a_1 &= e(\Psi(T_1)) = e((1,0)) = 1,\\
a_2 &= e(\Psi(T_2)) = e((0,1/2)) = x,\\
a_3 &= e(\Psi(T_3)) = e(\pi(T_3(v))) = e(\pi((0,-1,-1)^t))\\
&= e((-1,-1)^t) = -1-2x,\\
a_4 &= e(\Psi(T_4)) = e(\pi((0,-1,-1/2)^t)) = e((-1,-1/2)^t) = -1-x,\\
a_5 &= e(\Psi(T_5)) = e(\pi((0,0,1)^t)) = e((0,1)^t) = 2x,\\
a_{23} &= e(\Psi(T_{23})) = e(\pi((0,1,0)^t)) = e((1,0)^t) = 1,\\
a_{97} &= e(\Psi(T_{23})) = e(\pi((0,14,3)^t)) = e((14,3)^t) = 14+6x.
\end{aligned}$$

Example 2.20. It is easier to appreciate Algorithm 2.18 after seeing how big the coefficients of the power series expansion of a newform typically are, when the newform is defined over a large field. For example, there is a newform

$$f = \sum_{n=1}^{\infty} a_n q^n \in S_2(\Gamma_0(389))$$

such that if $\alpha = a_2$, then

$$\begin{aligned}
1097385680\cdot a_3(f) = &-20146763x^{19} + 102331615x^{18} + 479539092x^{17}\\
&- 3014444212x^{16} - 3813583550x^{15} + 36114755350x^{14}\\
&+ 6349339639x^{13} - 227515736964x^{12} + 71555185319x^{11}\\
&+ 816654992625x^{10} - 446376673498x^9 - 1698789732650x^8\\
&+ 1063778499268x^7 + 1996558922610x^6 - 1167579836501x^5\\
&- 1238356001958x^4 + 523532113822x^3 + 352838824320x^2\\
&- 58584308844x - 25674258672.
\end{aligned}$$

In contrast, if we take $v = \{0,\infty\} = (0,1) \in \boldsymbol{\mathcal{M}}_2(\Gamma_0(389))^+$, then

$$T_3(v) = -4(1,0) + 2(1,291) - 2(1,294) - 2(1,310) + 2(1,313) + 2(1,383).$$

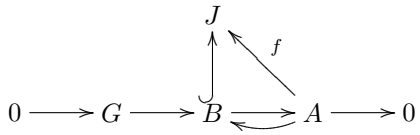
Storing $T_3(v), T_5(v), \ldots$ as vectors is more compact than storing $a_3(f)$, $a_5(f)$, $\ldots$ directly as polynomials in a_2!

3 Explicit Modular Abelian Varieties

3.1 Explicit Defining Data for Modular Abelian Varieties

We represent modular abelian varieties over $\mathbb{Q}$ *explicitly* as follows. Let J be an arbitrary finite product of modular Jacobians $J_0(N)$ for various N. (More generally, one would consider $J_H(N) = \mathrm{Jac}(X_H(N))$ for subgroups $H \subset (\mathbb{Z}/N\mathbb{Z})^*$.)

We will refer to J as an *ambient modular abelian variety.* Fix a modular abelian variety A and a finite degree homomorphism $\varphi : A \to J$. Then there is an isogeny from the image B of A in J back to A whose kernel we denote by G, so A is isomorphic to B/G and $B \subset J$:



In other words we can represent any modular abelian variety by giving

$$G \subset B \subset J,$$

all defined over $\mathbb{Q}$. It remains to explain how we explicitly specify B and G.

We specify B as follows. The inclusion $B \hookrightarrow J$ induces an inclusion of rational homology $\mathbb{H}_1(B, \mathbb{Q}) \hookrightarrow \mathbb{H}_1(J, \mathbb{Q})$ and B is determined by the image V of $\mathbb{H}_1(B, \mathbb{Q})$ in the $\mathbb{Q}$-vector space $\mathbb{H}_1(J, \mathbb{Q})$. We explicitly compute a basis for $\mathbb{H}_1(J, \mathbb{Z})$ and $\mathbb{H}_1(J, \mathbb{Q}) = \mathbb{H}_1(J, \mathbb{Z}) \otimes \mathbb{Q}$ using modular symbols, and specify B by giving a basis in reduced echelon form for a subspace $V \subset \mathbb{H}_1(J, \mathbb{Q})$. Of course, not every subspace corresponds to a modular abelian variety, but we can determine whether or not a given V corresponds to a valid abelian subvariety (see **??**).

We specify G as follows. Suppose V defines an abelian subvariety B of J as above. By the Abel-Jacobi theorem, we have

$$J(\mathbb{C}) \cong \mathbb{H}_1(J, \mathbb{R})/\mathbb{H}_1(J, \mathbb{Z}),$$

and letting $\Lambda = \mathbb{H}_1(J, \mathbb{Z}) \cap V$ we have $B(\mathbb{C}) \cong (V \otimes \mathbb{R})/\Lambda$. In particular,

$$B(\mathbb{C})_{\mathrm{tor}} \cong V/\Lambda,$$

and we specify $G \subset B(\mathbb{C})_{\mathrm{tor}}$ by giving the lattice L with $\Lambda \subset L \subset V$ such that $L/\Lambda \cong G$.

For brevity, henceforth we use the term *modular abelian variety* to mean a modular (or equivalently GL_2-type) abelian variety A that has been given explicitly by a triple (V, L, J) where $V \subset \mathbb{H}_1(J, \mathbb{Q})$, the lattice $L \subset V$ contains $\Lambda = V \cap \mathbb{H}_1(J, \mathbb{Z})$, and J is specified by a finite ordered list of congruence subgroups $\Gamma_0(N)$. (More generaly, one might include $\Gamma_1(N)$ and $\Gamma_H(N)$.) We use the notation (V, J) as a shorthand for $L = \Lambda$.

3.2 Endomorphism Rings and Hom Spaces

3.2.1 Computing End and Hom

The following saturation algorithm will be important when computing $\mathrm{End}(A)$ and $\mathrm{Hom}(A, B)$.

Algorithm 3.1 (Saturate)**.** Given a subgroup L of $\mathbb{Z}^n$, this algorithm computes the saturation $(\mathbb{Q}L)\cap\mathbb{Z}^n$ of L in $\mathbb{Z}^n$. Let M be a matrix whose rows are a $\mathbb{Z}$-basis for L.

1. [Hermite Normal Form] Find the Hermite Normal Form H of M^t.
2. [Inverse] Compute $S = (H^t)^{-1}M$ using the "last big row" trick. Then output S whose rows are a basis for the saturation of L.

Proof. It suffices to prove that $(H^t)^{-1}M$ has rows that span the saturation of the row span of M. [...] □

Note that one could instead replace H by an LLL reduced basis for the rowspace of M^t, but this is usually much slower because the p-adic/modular algorithm [[stein-pernet]] for computing Hermite normal form is fast.

If A is an abelian variety of dimension 2 then after chosing a basis for $\Lambda = \mathbb{H}_1(A,\mathbb{Z})$, we have

$$\mathrm{End}(\Lambda) \cong \mathrm{Mat}_{2d\times 2d}(\mathbb{Z}) \approx \mathbb{Z}^{(2d)^2}.$$

Proposition 3.2. *Let A be a simple abelian variety over a number field K, let $\Lambda = \mathbb{H}_1(A,\mathbb{Z})$ and embed $\mathrm{End}(A/K)$ in $\mathrm{End}(\Lambda)$ by the action of endomorphisms on homology. Then*

$$\mathrm{End}(A/K) = (\mathrm{End}(A/K)\otimes\mathbb{Q})\cap\mathrm{End}(\Lambda),$$

where the intersection takes place in $\mathrm{End}(\Lambda)\otimes\mathbb{Q}$.

We will use the following lemma in the proof of Proposition 3.2.

Lemma 3.3. *Let K be a number field. If an element $x\in\mathbb{C}$ is fixed by every element of $\mathrm{Aut}(\mathbb{C}/K)$, then $x\in K$.*

Proof. If $x\in\overline{K}$, this is standard Galois theory. If $x\notin\overline{K}$, then x is transcendental. Since $x+1$ is also transcendental, the fields $\overline{K}(x)$ and $\overline{K}(x+1)$ are isomorphic via a map σ sending x to $x+1$. Every automorphism of a subfield of $\mathbb{C}$ extends to $\mathbb{C}$, so σ extends to an automorphism of $\mathbb{C}$ that does not fix x. □

Proof of Proposition 3.2. An element of $\mathrm{End}(A/\mathbb{C})$ is just a complex linear map on $\mathrm{Tan}(A_{\mathbb{C}})$ that preserves Λ. The inclusion of $\mathrm{End}(A/K)$ in the right hand side is obvious, so suppose $\varphi\in(\mathrm{End}(A/K)\otimes\mathbb{Q})\cap\mathrm{End}(\Lambda)$. Then there is a positive integer n such that $n\varphi\in\mathrm{End}(A/K)$. Thus $n\varphi\in\mathrm{End}(A/K)\otimes\mathbb{Q}$ induces a complex-linear endomorphism of $\mathrm{Tan}(A_{\mathbb{C}})$, so $\varphi = (1/n)n\varphi$ also induces a complex-linear endomorphism of $\mathrm{Tan}(A_{\mathbb{C}})$; also, by hypothesis φ preserves Λ. Thus $\varphi\in\mathrm{End}(A/\mathbb{C})$.

There is a nonzero integer n such that $n\varphi$ is defined over K, so for any $\sigma\in\mathrm{Gal}(\mathbb{C}/K)$, we have $\sigma([n]\varphi)-[n]\varphi = 0$. But

$$\sigma([n]\varphi) = \sigma([n])\sigma(\varphi) = [n]\sigma(\varphi),$$

so

$$[n](\sigma(\varphi) - \varphi) = 0,$$

which implies $\sigma(\varphi) = \varphi$, since the kernel of $[n]$ is finite and the image of $\sigma(\varphi) - \varphi$ is either infinite or 0. By Lemma 3.3, $\varphi \in \mathrm{End}(A/K)$. □

Algorithm 3.4 (Endomorphism Algebra as Field)**.** Given a simple modular abelian variety A over $\mathbb{Q}$, this algorithm computes a number field F and an isomorphism $\mathrm{End}(A) \otimes \mathbb{Q} \to F$.

1. [Find A_f] Find an isogeny $\varphi : A \to A_f$, where A_f is a newform abelian variety.

2. [Choose random endomorphism] Randomly pick an endomorphism φ of A_f and compute its minimal polynomial g.

3. [Does endomorphism generate?] If $\deg g = \dim(A_f)$, then let F be the number field generated by a root α of g. Otherwise, go to step 1.

4. [Define an isomorphism] Let Ψ be the unique field homomorphism $\mathrm{End}(A_f) \otimes \mathbb{Q} \to F$ that sends φ to α. Compose this with the isomorphism $\mathrm{End}(A) \otimes \mathbb{Q} \to \mathrm{End}(A_f) \otimes \mathbb{Q}$ induced by φ to obtain the desired isomorphism.

Proof. By [**?**] because A is simple, modular, and defined over $\mathbb{Q}$, we know that $\mathrm{End}(A) \otimes \mathbb{Q}$ is a number field of degree equal to $\dim(A)$. (If we instead consider $\mathrm{End}(A/\overline{\mathbb{Q}})$, then $\mathrm{End}(A/\overline{\mathbb{Q}}) \otimes \mathbb{Q}$ could be a non-commutative division algebra. Again we emphasize that by definition $\mathrm{End}(A)$ contains only the endomorphisms of A that are defined over $\mathbb{Q}$.)

By the primitive element theorem, there exists a φ such that if f is the minimal polynomial of φ, then $\deg(f) = \dim(A)$. Then since $\deg(f) = \dim(A)$ it follows that the map Ψ is an isomorphism (a nonzero homomorphism between number fields of the same dimension is an isomorphism). □

Algorithm 3.5 (Compute $\mathrm{End}(A)$)**.** Given a simple modular abelian variety A, this algorithm computes $\mathrm{End}(A)$.

1. [Find Modular Form] Since A is simple we can use Algorithm **??** to find a newform f such that A is isogenous to the abelian variety A_f. It suffices to compute $\mathrm{End}(A) \otimes \mathbb{Q} = \mathrm{End}(A_f) \otimes \mathbb{Q}$, since by Proposition 3.2 this yields $\mathrm{End}(A)$. Thus it suffices to compute $\mathrm{End}(A_f)$.

2. [Initialize] Let $d = \dim(A_f)$, let $n = 1$, and let V be the zero subspace of $\mathrm{End}(A_f) \otimes \mathbb{Q}$.

3. [Compute Hecke operator] Using Algorithm **??**, compute the restriction of the Hecke operator T_n to A_f, as an element of $\mathrm{End}(A_f) \otimes \mathbb{Q}$.

4. [Increase V] Replace V by $V + \mathbb{Q} \cdot T_n$.

5. [Finished?] If $\dim(V) < d$, increase n and go to Step 3.

6. [Saturate] Compute $\mathrm{End}(A_f/\mathbb{Q}) = V \cap \mathrm{End}(\Lambda_{A_f})$ using Algorithm **??**.

Proof. We need to show that the algorithm terminates, i.e., that the Hecke algebra generates $\mathrm{End}(A_f/\mathbb{Q}) \otimes \mathbb{Q}$. But by [**?**, Thm. 1] the image of $T \otimes \mathbb{Q}$ in $\mathrm{End}(A_f/\mathbb{Q}) \otimes \mathbb{Q}$ is a subfield of degree $\dim A_f$. But A_f is simple by [**?**, Cor. 4.2], so [**?**, Thm. 2.1] implies that $\mathrm{End}(A_f/\mathbb{Q}) \otimes \mathbb{Q}$ also has dimension $\dim(A_f)$. Thus the Hecke algebra generates $\mathrm{End}(A_f/\mathbb{Q}) \otimes \mathbb{Q}$. By Proposition 3.2 once we have $\mathrm{End}(A_f/\mathbb{Q}) \otimes \mathbb{Q}$ we apply Algorithm **??** to get $\mathrm{End}(A_f/\mathbb{Q})$. □

Algorithm 3.6 (Compute $\mathrm{Hom}(A, B)$)**.** Given modular abelian varieties A and B, we compute $\mathrm{Hom}(A, B)$ as follows.

1. [Factorizations] By Proposition 3.2 it suffices to explain how to compute $\mathrm{Hom}(A, B) \otimes \mathbb{Q}$. For this, we compute using Algorithm **??** factorizations $\prod_{i\in I} C_i^{e_i}$ and $\prod_{i\in I} C_i^{f_i}$ of A and B up to isogeny (with isogenies) respectively, where I is some index set, the $C_i's$ are non-isogenous simple abelian varieties, and $e_i, f_i \geq 0$. For the rest of this algorithm we replace A, B, by these products.

2. [Simple case] When $A \sim C^e$ and $B \sim D^f$, where C, D are simple abelian varieties we compute $\mathrm{Hom}(A, B)$ in the following way. If C and D are not isogenous $\mathrm{Hom}(A, B) = 0$. If C and D are isogenous,

$$\mathrm{Hom}(A, B) \otimes \mathbb{Q} = \mathrm{Hom}(C^e, D^f) \otimes \mathbb{Q} = \mathrm{Mat}_{e\times f}(\mathrm{End}(C) \otimes \mathbb{Q}).$$

3. [General case] We compute each $\mathrm{Hom}(C_i^{e_i}, C_j^{f_j}) \otimes \mathbb{Q}$ as in Step 2 and obtain $\mathrm{Hom}(\prod C_i^{e_i}, \prod C_j^{f_j}) \otimes \mathbb{Q}$ as a matrix with blocks $\mathrm{Hom}(C_i^{e_i}, C_j^{f_j}) \otimes \mathbb{Q}$ for each pair (i, j).

Proof. Suppose first that $A \sim C^e$, $B \sim D^f$ with C, D simple abelian varieties. When C and D are not isogenous there is no morphism $A \to B$, so $\mathrm{Hom}(A, B) = 0$. When C and D are isogenous, a morphism $C^e \to D^f$ over $\mathbb{Q}$ is given by an $e \times f$ matrix with entries from $\mathrm{End}(A) \otimes \mathbb{Q}$, where the (i, j)th entry represents the morphism between the ith component of A and jth component of B. We get $\mathrm{End}(A) \otimes \mathbb{Q}$ using Algorithm **??**. Once we have $\mathrm{Hom}(A, B) \otimes \mathbb{Q}$, to get $\mathrm{Hom}(A, B)$ we only need to apply Proposition 3.2.

In general, when $A = \prod_{i\in I} C_i^{e_i}$ and $B = \prod_{i\in I} C_i^{f_i}$ we get $\mathrm{Hom}(C_i^{e_i}, C_j^{f_j})$ as before and combining these blocks we obtain $\mathrm{Hom}(A, B)$. □

3.3 Isogenies and Isomorphisms Between Modular Abelian Varieties

3.3.1 Isogenies From A to B

Algorithm 3.7 (Test if Isogenous)**.** Given two modular abelian varieties A and B, this algorithm decides whether or not A and B are isogenous, and if so returns an isogeny between them.

1. [A, B both simple] When A and B are both simple they are isogenenous to abelian varieties A_f and A_g attached to newforms; we can find explicit isogenies using Algorithm **??**. Then A is isogenous to B if and only if $A_f = A_g$, i.e., f and g are Galois conjugate.

2. [Pair off factors] When A and B are not simple we pair off factors, i.e. for any C in a factorization of A we check if there is an isogenous D in a factorization of B. If such D exists and the multiplicities of C in A and D in B are the same we remove D and continue with another C. Otherwise, A and B cannot be isogenous.

Proof. When A and B are simple, by [**?**, §5] $A \simeq A_f$ and $B \simeq A_g$ are isogenous if and only if the corresponding newforms f and g are Galois conjugate, since f and g determine $L(A_f, s)$ ad $L(A_g, s)$.

If $A \sim \prod_{i\in I} A_i^{e_i}$ and $B \sim \prod_{i\in I} B_i^{e_i}$, indexed so that $A_i \sim B_i$ for all $i \in I$, then we get that the products $\prod_{i\in I} A_i^{e_i}$ and $\prod_{i\in I} B_i^{e_i}$ are isogenous, so A and B are also isogenous.

Conversely, suppose that $A \sim B$ and $\varphi : A \to B$ is some isogeny. Let $A \sim \prod_{i\in I} A_i^{e_i}$ and $B \sim \prod_{j\in J} B_j^{f_j}$ be factorizations of A and B into products of powers of non-isogenous simple abelian varieties. Fix an index $i \in I$. Combining the maps from A_i to A, from A to B, and the projection to B_j for each j we obtain morphisms $\phi_{ij} : A_i \to B_j$ for all $j \in J$. Since the image of an abelian variety is an abelian variety and all B_j's are simple it follows that $\varphi_{ij}(A_i)$ is either zero or all of B_j, which means that A_i and B_j are isogenous. It is not possible that all $\varphi_{ij}(A_i)$ are zero since that would imply that φ is the zero map, so we find a B_j isogenous to A_i. Removing A_i and B_j from the factorizations and repeating this argument yields that A and B are isogenous if and only if there is a bijection $\sigma : I \to J$ such that A_i is isogenous to $B_{\sigma(i)}$ for all i, and $e_i = f_{\sigma(i)}$. □

3.3.2 Isomorphisms from A to B

In this section we describe an algorithm to decide whether two simple modular abelian varieties are isomorphic, and if so to give an isomorphism. We do not yet know an algorithm to decide whether two nonsimple modular abelian varieties are isomorphic (just need a way to enumerate elements in lattice of small norm – might be straightforward if don't care about speed!).

Algorithm 3.8 (Norm Equation)**.** Given an order $\mathcal{O}$ in a number field K and an element $a \in \mathbb{Q}$, this algorithm finds all solutions in $\mathcal{O}$ to the norm equation $\mathrm{Norm}(x) = a$, up to units of $\mathcal{O}$.

Replace the following by a reference to Henri Cohen's book, etc. [[Claus Fieker suggests the following algorithm (we should expand on that)

1. [Class Group] Find the class group of K.

2. [Ideals of bounded norm] Use linear programming [[huh??]] to find all ideals of norm up to some bound.

3. [Solve] Deduce all solutions to the norm equation up to units.

]]

Algorithm 3.9 (Test if Isomorphic)**.** Given simple modular abelian varieties A and B, this algorithm either proves that A and B are not isomorphic, or returns an isomorphism between them (or all isomorphisms, up to units).

1. [Equal?] If $A = B$, return "yes" and the identity map.
2. [Isogenous?] Determine whether A and B are isogenous using Algorithm **??**. If A and B are not isogenous then return "no", and if A and B are isogenous, let $f : B \to A$ be an isogeny.
3. [Degree of isogeny] Compute $d = \deg(f)$. If d is not a square, return "no".
4. [Endomorphism algebra] Compute the number field $K = \mathrm{End}(A) \otimes \mathbb{Q}$, and an embedding of $\mathrm{End}(A)$ into K using Algorithm **??**.
5. [Hom space] Compute $\mathrm{Hom}(A, B)$ using Algorithm **??**.
6. [Image of Hom space] Compute the image H_f of $\mathrm{Hom}(A, B)$ in $\mathrm{End}(A)$ got by composing with f.
7. [Endomorphism ring] Compute the order $\mathcal{O}$ in K equal to $\mathrm{End}(A)$ using Algorithm **??**.
8. [Solve norm equation] Find solutions (up to units of $\mathcal{O}$) of the norm equations $\mathrm{Norm}(x) = \pm\sqrt{d}$ in $\mathcal{O}$. If there are no solutions, return "no".
9. [Lift to H_f?] For each solution (up to units), check whether it lies in H_f.
10. [Isomorphic?] If a solution x lies in H_f, then return "yes" and $x \circ f^{-1}$. (Note that at this point we could also output $x \circ f^{-1}$ and continue on to return representatives for all isomorphisms up to units.)
11. [Not isomorphic?] If none of the solutions lies in H_f, return "no".

Proof. Let $f : B \to A$ be an isogeny and denote its degree by d. Define

$$H_f = \{f \circ g : g \in \mathrm{Hom}(A, B)\} \subset \mathrm{End}(A).$$

Since degree is multiplicative, A and B are isomorphic if and only if the subset H_f of $\mathrm{End}(A)$ contains an element of degree d. Embed $\mathrm{End}(A)$ into the number field $K = \mathrm{End}(A) \otimes \mathbb{Q}$ and let $\mathcal{O}$ be the order in K that is the image of $\mathrm{End}(A)$. By [**?**, Prop 12.12], for $x \in K$ we have $\mathrm{Norm}(x)^2 = \deg(x)$. Thus, finding an element of degree d in H_f is equivalent to finding $x \in \mathcal{O}$ with $\mathrm{Norm}(x) = \pm\sqrt{d}$, such that $x \in H_f$, where we view H_f as a subset of K using the above inclusions.

Using Algorithm **??**, we find all x such that $\mathrm{Norm}(x) = \pm\sqrt{d}$, up to units of $\mathcal{O}$. There are may be infinitely many units, e.g., if K is a real quadratic field, so there are often infinitely many solutions to the norm equation and we cannot

directly check whether at least one of these infinitely many are in H_f. However, because there are only finitely many solutions up to units, it will suffice to show that H_f is stable under units and to check whether each representative solution is in H_f. Thus to finish the proof of correctness of the algorithm, we verify that $x \in H_f$ if and only if $xu \in H_f$, where u is any unit of $\mathcal{O}$. If $x = f \circ g$ for some $g \in \mathrm{Hom}(A, B)$, then $xu = f \circ (g \circ u)$ is in H_f since $g \circ u \in \mathrm{Hom}(A, B)$. Conversely, if $xu \in H_f$, then by what we have just shown $x = xuu^{-1} \in H_f$. $\square$

Discuss how non-simple case works. Still just need to solve a norm equation but solving it is more complicated (?).

3.3.3 The Minimal Isogeny

A small extension of Algorithm **??** gives us the minimal degree of any isogeny between two isogenous modular abelian varieties. [[delete below and just say that we run through all square multiples of d instead of just d in the algorithm above. the below is riddled with errors anyways.]]

Algorithm 3.10 (Minimal Isogeny)**.** Given simple modular abelian varieties A and B, this algorithm checks if A and B are isogenous and if so returns the minimal degree of an isogeny $A \to B$ together with an isogeny of that degree.

1. [Equal?] If $A = B$, return 1 and the identity map.
2. [Isogenous?] Determine whether A and B are isogenous using Algorithm **??**. If A and B are not isogenous then return "not isogenous", and if A and B are isogenous, let $f : B \to A$ be some isogeny.
3. [Degree of some isogeny] Compute $\deg(f)$ using Algorithm **??**. Write $\deg(f)$ as ab^2, where a is squarefree.
4. [Endomorphism algebra] Compute the number field $K = \mathrm{End}(A) \otimes \mathbb{Q}$, and an embedding of $\mathrm{End}(A)$ into K using Algorithm **??**.
5. [Hom space] Compute $\mathrm{Hom}(A, B)$ using Algorithm **??**.
6. [Image of Hom space] Compute the image H_f of $\mathrm{Hom}(A, B)$ in $\mathrm{End}(A)$ got by composing with f...
7. [Endomorphism ring] Compute the order $\mathcal{O}$ in K generated by $\mathrm{End}(A)$ Algorithm **??**.
8. [Initialize] Let $i = 0$.
9. [Solve norm equation] Increase i by one and find the solutions (up to units of $\mathcal{O}$) of the norm equations $\mathrm{Norm}(x) = \pm abi$ in $\mathcal{O}$. If there are no solutions, repeat this step.
10. [Lift to H_f?] For each solution (up to units), check whether it lies in H_f.

11. [Isogenous of degree ai^2?] If a solution x lies in H_f, then return ai^2 and $x \circ f^{-1}$.

12. [Should try isogeny of higher degree] If none of the solutions lies in H_f, return to Step 9.

Proof. Let $f : A \to B$ be an isogeny and denote its degree by $d = ab^2$, where a is squarefree. Define $H_f = \{\phi \circ f : \phi \in \mathrm{Hom}(B, A)\} \subset \mathrm{End}(A)$. Since degree is multiplicative, B and A are isogenous via an isogeny of degree d' if and only if H_f contains an element of degree dd'. Embed $\mathrm{End}(A)$ into $K = \mathrm{End}(A) \otimes \mathbb{Q}$ and let $\mathcal{O}$ be the order in K generated by $\mathrm{End}(A)$. By Proposition 12.12. in Milne's "Abelian Varieties" for $x \in K$ we have $\mathrm{Norm}^2(x) = \deg(x)$. Thus, finding an element of degree dd' in H_f is equivalent to finding $x \in \mathcal{O}$ with $\mathrm{Norm}(x) = \pm\sqrt{dd'}$, such that x actually comes from H_f. Hence, the possible values for d' are ai^2 for $i \in \mathcal{N}$. We can find all x such that $\mathrm{Norm}(x) = \pm\sqrt{dd'}$ up to units of $\mathcal{O}$. The proof that this suffices is the same as the end of the proof of Algorithm 3.9. □

4 Quaternion Algebras

4.1 Basic Facts about Quaternion Algebras over Number Fields

These notes are just a short overview of some basic facts about quaternion algebras over $\mathbb{Q}$ and over other number fields, which end by stating the classification of quaternion algebras over a number field.

4.1.1 Hamilton's quaternions

I suspect everyone reading these notes has heard of the Hamiltonian quaternions, usually denoted $\mathbb{H}$. This is just the set

$$\mathbb{H} = \{a + bi + cj + dk \mid a, b, c, d \in \mathbb{Q}\},$$

with addition defined as usual, and multiplication defined by the rules

$$i^2 = -1, \quad j^2 = -1, \quad \text{and} \quad ij = k = -ji.$$

(Maybe it's more common to choose $\mathbb{R}$ instead of $\mathbb{Q}$, but since we're going to be thinking about number fields, we'll stick with $\mathbb{Q}$.) The first reason most people run into this ring is because it's an example of a *division ring* or *skew field*, i.e. a non-commutative field. Indeed, given a non-zero quaternion $x = a+bi+cj+dk$, we have that for

$$x^{-1} = \frac{a - bi - cj - dk}{\sqrt{a^2 + b^2 + c^2 + d^2}},$$

in fact $xx^{-1} = x^{-1}x = 1$.

The Hamiltonian quaternions pop up all over the place in math, but also in physics. If I knew any physics, maybe I'd know why?

4.1.2 Quaternion Algebras

Now as you probably guessed, I'm not going to spend the whole hour just talking about $\mathbb{H}$. In fact, there are lots of things similar to $\mathbb{H}$ that are just as interesting – and they all come from the same construction as above, but replacing -1 by other choices.

Also, I'm going to talk about quaternion algebras over number fields other than $\mathbb{Q}$. However, there's really no loss in just talking about the situation over $\mathbb{Q}$, if that's more comfortable.

*L*et F be a field. A **quaternion algebra (over F)** is a 4-dimensional central simple algebra over F. That is, it is a 4-dimensional vector space over F which is also a ring, with center isomorphic to F, and which has no nontrivial two-sided ideals.

This means that B consists of elements of the form $\alpha = x + yi + zj + wij$, with the multiplication rules as above. We define the **conjugate** of α, $\overline{\alpha}$, to be the quaternion $x - yi - zj - wij$. We also define the **reduced trace**

$$\mathrm{tr}(x) = \alpha + \overline{\alpha} = 2x$$

and the **reduced norm**

$$\mathrm{nm}(x) = \alpha\overline{\alpha} = x^2 - ay^2 - bz^2 + abw^2.$$

We won't do much with these right now, but they're important in the study of quaternion algebras. For instance, α is invertible if and only if it has reduced norm not equal to 0, and the elements of reduced trace 0 form a useful subspace of B. (They're the elements of B which are not in K, but whose square is in K. They're like the purely imaginary elements in $\mathbb{C}$.)

So clearly $\mathbb{H}$ above is going to be an example; here's another easy one: $\mathcal{M}_2(F)$, the ring of 2×2 matrices over F, is always a quaternion algebra over F. Now, this matrix example just feels like cheating – we somehow want to think of that as a "degenerate" example of a quaternion algebra. Let's come up with some terminology to do just that.

*L*et F be a field, and B a quaternion algebra over F. Let K be any extension of F (or, in fact, any field which is an F-algebra). We say that B **splits over** K if $B \otimes_F K \cong \mathcal{M}_2(F)$. Otherwise, we say that B is **ramified over** K. If $K = F$, we simply say B is split or ramified.

So now you probably want some more exciting examples of quaternion algebras than just these two, which I've hinted at above. Let $a, b \in F^\times$. Then we can define a quaternion algebra B as the 4-dimensional algebra over F on basis $\{1, i, j, ij\}$ with

$$i^2 = a, \quad j^2 = b, \quad ij = -ji.$$

You can check that this does indeed give us a quaternion algebra (i.e. that it's central and simple over F). We'll denote this quaternion algebra by the somewhat heavy notation

$$B = \mathrm{quatalg}(a, b, F),$$

which will make at least a bit more sense shortly.

Several natural questions should pop into your head, such as:

- Does every quaternion algebra look like this? (Yes, unless F has characteristic 2 – but then it's your fault for working over a field of characteristic 2.)
- Can I easily tell when two of these are isomorphic? (Yes.)
- Are these easily parametrized over $\mathbb{Q}$, or any number field? (No ... just kidding. Yes.)

In fact, it turns out that over any number field F, we can associate to B a *discriminant* which completely determines B up to isomorphism.

Let F be a number field, and let M_F denote the set of places of F. Recall that this consists of an embedding for each equivalence class of norms on F; by Ostrowski's Theorem, we get one for each prime in the ring of integers of F, one for each real embedding of F, and one for each complex conjugate pair of embeddings of F. Given a place $v \in M_F$, let $\mathbb{F}_v$ denote the completion of F at v, and let $B_v \cong B \otimes_F F_v$.

If your algebraic number theory is a little rusty, that's okay – just take $F = \mathbb{Q}$ below. You really don't lose any of the content.

It turns out that B will be completely determined (up to isomorphism) by the B_v for all v. This means that we should start by asking what the possibilities for quaternion algebras over $\mathbb{R}$, $\mathbb{C}$, and over the finite extensions of $\mathbb{Q}_p$, i.e. over local fields. Needless to say, there's a simple classification:

Theorem 4.1. *Over any local field, there is only one ramified quaternion algebra up to isomorphism.*

This also applies over any infinite places. We say that B is **definite** if it is ramified at *every* infinite place of F, and **indefinite** otherwise. (Some authors only define this notion in the case where F is totally real.) So now we want to start putting this together to try to determine B:

Definition 4.2. Let B be the quaternion algebra $\text{quatalg}(a, b, F)$, and v a place of F. We define the **Hilbert symbol** $(a, b)_v$ to be 1 if B is split over F_v, and -1 if B is ramified over F_v.

Now I'm just going to state a string of theorems about the Hilbert symbol; these are easily proven by hand, or you can look in Serre's *A Course in Arithmetic* [[**?**]]. (In fact, if you haven't already, you should read that book from cover to cover.) We then have the following theorems:

Theorem 4.3. *We have that $(a, b)_v$ is 1 exactly when the quadratic form $z^2 - ax^2 - by^2$ has a nontrivial solution over F_v (i.e., a solution where x, y, and z are not all zero).*

Theorem 4.4. *If a is a square in F_v, then $(a,b)_v = 1$. (Simply take z to be the square root of a, $x = 1$, $y = 0$. Of course, the same applies to b, mutatis mutandis.)*

Theorem 4.5. *If a and b are both squares in $\mathbb{F}_v^\times$, then $(a,b)_v = 1$.*

Now, we know that any given a, b in F are going to be units in F_v for almost every v. Then the previous theorem says that B is going to be split at almost every place! We have even more, in fact:

Theorem 4.6. *(Hilbert) We have that*

$$\prod_{v \in M_F} (a,b)_v = 1.$$

So let's summarize what the previous theorems just said. Given a quaternion algebra B over F, we know that B is ramified at a finite, even number of places of F, and is split elsewhere. So we define the **discriminant** of B, $\mathrm{disc}(B)$, to be the product of the places where B is ramified. (One can think of this as a (squarefree) ideal of F, along with a collection of some of the infinite places of F.) Now, of course, we want to know: does this determine B? Yes!

Theorem 4.7. *Let S be a finite, even cardinality set of places in M_F. Then there exists a unique quaternion algebra over F which is ramified exactly at S up to isomorphism.*

4.1.3 Bibliography

There are several good sources to learn about quaternion algebras and related topics. For quaternion algebras themselves, it's hard to beat Vigneras's [[**?**]]. Another book along the same lines which seems nice (though I haven't read much), and has the advantage of at least claiming a computational bent, is [[**?**]]. For information about Shimura curves, it's hard to beat Shimura himself in [[**?**]] ("Read the masters!"). One can also easily find the original papers by Shimura.

4.2 Quaternion Algebras and Supersingular Elliptic Curves

The main reference for this section is [[Kohel, Hecke module structure on quaternions]].

4.2.1 Eichler Orders and Supersingular Curves

Let p be a prime number. Recall that up to isomorphism there is a unique quaternion algebra H that is ramified precisely at p and ∞. An *order* R in H is a subring containing a $\mathbb{Q}$-basis for H that is finitely generated as a $\mathbb{Z}$-module.

Unlike the situation with orders of number fields, maximal orders in quaternion algebras are *not* unique. Indeed, the conjugate of any maximal order is also maximal.

An *Eichler order* R in H is the intersection of two distinct maximal orders in H. The *level* of R is the index of R in any maximal order that contains R.

A *supersingular elliptic curve* E over $\mathbb{F}_p$ is a curve such that $E(\overline{\mathbb{F}}_p)[p] = 0$. Let M be in integer coprime to p. An *enhanced elliptic curve* $\mathbf{E} = (E, C)$ is a pair consisting of an elliptic curve E over $\overline{\mathbb{F}}_p$ and a cyclic subgroup $C \subset E(\overline{\mathbb{F}}_p)$ of order M. If $\mathbf{E} = (E, C)$ and $\mathbf{E}' = (E', C')$ are enhanced curves, then a *morphism* $\mathbf{E} \to \mathbf{E}'$ is a homomorphism $E \to E'$ that sends C into C'. We say $\mathbf{E}$ is supersingular if E is supersingular.

Theorem 4.8. *The enhanced curve* $\mathbf{E}$ *is supersingular if and only if the endomorphism ring* $\mathrm{End}(\mathbf{E})$ *is an Eichler order of level* M *in the quaternion algebra ramified at* p *and* ∞.

See [[Silverman AEC I, §V.3]] for most of the proof. The basic idea is that if $\mathrm{End}(\mathbf{E})$ isn't a quaternion order, then it is an order in a number field, and using properties of the Frobenius endomorphism, one concludes that E isn't supersingular, and conversely.

4.2.2 The Supersingular Module

Let $\boldsymbol{\mathcal{S}} = X_0(Mp)^{\mathrm{ss}}_{\overline{\mathbb{F}}_p}$ denote the set of isomorphism classes of enhanced supersingular elliptic curves $\mathbf{E}$. Let $X = \mathrm{Div}(X_0(Mp)^{\mathrm{ss}}_{\overline{\mathbb{F}}_p})$ be the free abelian group on the elements of $\boldsymbol{\mathcal{S}}$.

The property of being supersingular is preserved under isogeny, which allows us to define an action of Hecke operators on X. For $n \not| Mp$, the Hecke operators T_n act on X by

$$T_n([\mathbf{E}]) = \sum_{\varphi} [\mathbf{F}],$$

where the sum is over the cyclic isogenies $\varphi : \mathbf{E} \to \mathbf{F}$ of degree n. There is also an inner product on X given by extending the following inner product linearly:

$$\langle [\mathbf{E}], [\mathbf{F}] \rangle = \frac{1}{2} \#\mathrm{Isom}(\mathbf{E}, \mathbf{F}),$$

and we have

$$\langle [\mathbf{E}], T_n([\mathbf{F}]) \rangle = \langle T_n([\mathbf{E}]), [\mathbf{F}] \rangle.$$

When $M = 1$, it is straightforward to directly compute with X, using what is called the *Mestre Method of Graphs*. We represent $\mathbf{E}$ in X by its supersingular j-invariants $j(E) \in \mathbb{F}_{p^2}$. Because the Hecke operators are algebraic correspondence, there is an explicit polynomial $\Phi_n(Z, W) \in \mathbb{Q}[Z, W]$ that we can use to compute the action of Hecke operators T_n. In particular, if $\mathbf{E} \in X$ has j-invarniant j, then $T_n([j]) = \sum_{j'}[j']$, where the sum is over the roots $j' \in \mathbb{F}_{p^2}$ of $\Phi_n(j, W)$. For example, the modular polynomial Φ_2 is

$$\Phi_2(Z, W) = -W^2Z^2 + W^3 + 1488\,W^2Z + 1488\,WZ^2 - 162000\,W^2 + Z^3 +$$

$$40773375\,WZ - 162000\,Z^2 + 8748000000\,W + 8748000000\,Z - 157464000000000$$

When $M > 1$, it is much more difficult to directly compute with the module X on supersingular enhanced curves. In fact, I know of no direct implementation, except when $X_0(M)$ has genus 0, when there is also an easy way to compute X. Fortunately, quaternion algebras come to the rescue and provide an *indirect* way to compute with X in general.

4.2.3 An Equivalence of Categories

Let $\mathcal{E}ll$ be the category of enhanced supersingular elliptic curves over $\overline{\mathbb{F}}_p$, fix an object $\mathbf{E}$ in $\mathcal{E}ll$, and set $R = \mathrm{End}(\mathbf{E})$. Recall Theorem 4.8, which asserts that R is an Eichler order the quaternion algebra ramified at p and ∞. Let Mod_R be the cateogry of locally free rank 1 right R-modules.

Theorem 4.9. *The map* $\mathbf{F} \mapsto \mathrm{Hom}(\mathbf{E}, \mathbf{F})$ *induces a functorial equivalence of categories* $\mathcal{E}ll \to \mathrm{Mod}_R$.

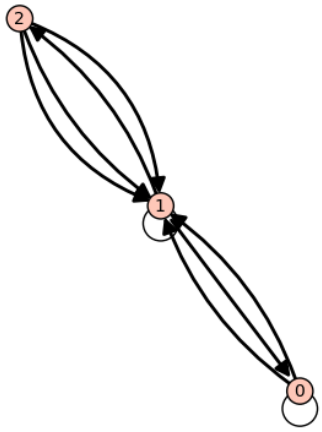
Thus the theorem implies that there is a natural bijection between the elements of $\boldsymbol{\mathcal{S}}$ and the nonzero right ideal classes Cl_R in R. Moreover, the action of Hecke operators on $\boldsymbol{\mathcal{S}}$ carries over to an action of Hecke operators on the free abelian group on the elements of Cl_R.

4.2.4 Example

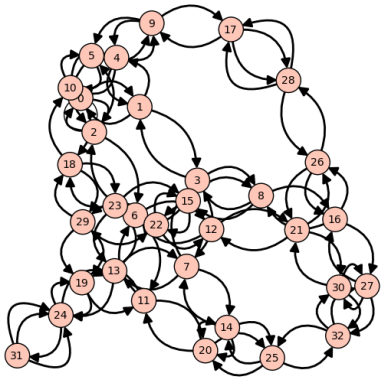
Example 4.10. We compute a supersingular j-invariant in characteristic 23, then find the isogenous j-invariants.

```
sage: k.<a> = GF(23^2)
sage: j = supersingular_j(k); j
3
sage: R.<Z> = k[]; S.<W> = R[]
sage: phi = sage.modular.ssmod.ssmod.Phi_polys(2,W,Z)
sage: phi(j,Z)
Z^3 + 5*Z^2 + 15*Z + 21
sage: phi(j,Z).roots()
[(3, 1), (19, 2)]
```

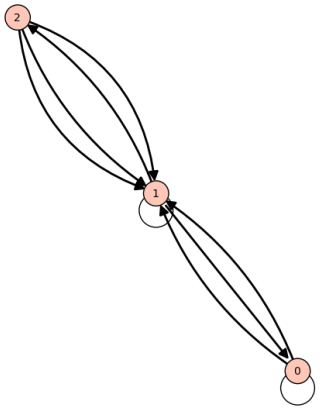
```
sage: X = SupersingularModule(23)
sage: X.supersingular_points()
([3, 19, 0], {19: 1, 0: 2, 3: 0})
sage: t2 = X.hecke_matrix(2); t2
[1 2 0]
[1 1 1]
[0 3 0]
sage: t2.fcp()
(x - 3) * (x^2 + x - 1)
sage: G = DiGraph(t2); show(G)
```




```
sage: X = SupersingularModule(389)
sage: X.supersingular_points()
([220, 46*a + 308, 343*a + 379, 85*a + 350, 24*a + 317, 365*a + 168, 304*a + 33,
  241*a + 74, 315*a + 162, 154, 17, 148*a + 150, 74*a + 200, 196*a + 187, 71*a + 182,
  92*a + 276, 290*a + 184, 36, 358, 193*a + 202, 318*a + 114, 99*a + 361, 297*a + 29,
  250*a + 201, 318, 7, 121, 71*a + 207, 327, 139*a + 367, 318*a + 139, 0, 16],
  {0: 31, 318*a + 114: 20, 74*a + 200: 12, 7: 25, 71*a + 207: 27, 16: 32, 17: 10,
  315*a + 162: 8, 154: 9, 365*a + 168: 5, 318*a + 139: 30, 36: 17, 343*a + 379: 2,
  148*a + 150: 11, 139*a + 367: 29, 193*a + 202: 19, 24*a + 317: 4, 85*a + 350: 3,
  318: 24, 196*a + 187: 13, 99*a + 361: 21, 327: 28, 290*a + 184: 16,
  71*a + 182: 14, 220: 0, 241*a + 74: 7, 358: 18, 297*a + 29: 22, 92*a + 276: 15,
  304*a + 33: 6, 46*a + 308: 1, 250*a + 201: 23, 121: 26})
sage: t2 = X.hecke_matrix(2); t2
33 x 33 sparse matrix over Integer Ring
sage: G = DiGraph(t2); G.plot(figsize=8)
```

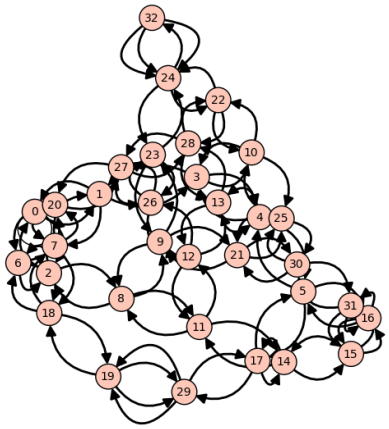



```
sage: B = BrandtModule(23); B
Brandt module of dimension 3 of level 23 of weight 2 over Rational Field
sage: B.order_of_level_N()
Order of Quaternion Algebra (-1, -23) with base ring Rational Field with basis (1/2 + 1/2*j, 1/2*i + 1
sage: B.right_ideals()
(Fractional ideal (2 + 2*j, 2*i + 2*k, 4*j, 4*k),
 Fractional ideal (2 + 2*j, 2*i + 6*k, 8*j, 8*k),
 Fractional ideal (2 + 10*j + 8*k, 2*i + 8*j + 6*k, 16*j, 16*k))
sage: t2 = B.hecke_matrix(2); t2
[1 2 0]
[1 1 1]
[0 3 0]
sage: DiGraph(t2).plot()
```




```
sage: B = BrandtModule(23,11); B
Brandt module of dimension 22 of level 23*11 of weight 2 over Rational Field
sage: B.hecke_operator(2).charpoly().factor()
```

```
(x - 3) * (x^2 + x - 1)^2 * (x^3 - 3*x^2 + 3) * (x^3 + x^2 - 4*x + 1) *
   (x^5 + 4*x^4 - 14*x^2 - 13*x - 1) *
   (x^6 - 3*x^5 - 4*x^4 + 16*x^3 - 3*x^2 - 10*x + 1)
```




```
sage: B = BrandtModule(389); B
Brandt module of dimension 33 of level 389 of weight 2 over Rational Field
sage: for I in B.right_ideals(): print I
Fractional ideal (2 + 2*j + 2*k, i + 2*j + k, 4*j, 4*k)
Fractional ideal (2 + 6*j + 2*k, i + 2*j + k, 8*j, 8*k)
Fractional ideal (2 + 6*j + 6*k, i + 6*j + k, 8*j, 8*k)
Fractional ideal (2 + 14*j + 2*k, i + 2*j + 9*k, 16*j, 16*k)
Fractional ideal (2 + 14*j + 2*k, i + 2*j + 25*k, 32*j, 32*k)
Fractional ideal (2 + 14*j + 2*k, i + 2*j + 57*k, 64*j, 64*k)
Fractional ideal (2 + 14*j + 6*k, i + 6*j + k, 16*j, 16*k)
Fractional ideal (2 + 14*j + 10*k, i + 10*j + k, 16*j, 16*k)
Fractional ideal (2 + 14*j + 14*k, i + 14*j + 9*k, 16*j, 16*k)
Fractional ideal (2 + 14*j + 14*k, i + 14*j + 9*k, 32*j, 32*k)
Fractional ideal (2 + 14*j + 18*k, i + 18*j + 9*k, 32*j, 32*k)
Fractional ideal (2 + 14*j + 30*k, i + 30*j + 25*k, 32*j, 32*k)
Fractional ideal (2 + 14*j + 30*k, i + 30*j + 25*k, 64*j, 64*k)
Fractional ideal (2 + 14*j + 34*k, i + 34*j + 25*k, 64*j, 64*k)
Fractional ideal (2 + 14*j + 62*k, i + 62*j + 57*k, 64*j, 64*k)
Fractional ideal (2 + 14*j + 62*k, i + 62*j + 121*k, 128*j, 128*k)
Fractional ideal (2 + 14*j + 66*k, i + 66*j + 121*k, 128*j, 128*k)
Fractional ideal (2 + 14*j + 126*k, i + 126*j + 57*k, 128*j, 128*k)
Fractional ideal (2 + 30*j + 6*k, i + 6*j + 17*k, 32*j, 32*k)
Fractional ideal (2 + 30*j + 6*k, i + 6*j + 17*k, 64*j, 64*k)
Fractional ideal (2 + 30*j + 22*k, i + 22*j + k, 32*j, 32*k)
Fractional ideal (2 + 46*j + 14*k, i + 14*j + 9*k, 64*j, 64*k)
Fractional ideal (2 + 46*j + 18*k, i + 18*j + 41*k, 64*j, 64*k)
Fractional ideal (2 + 46*j + 46*k, i + 46*j + 41*k, 64*j, 64*k)
Fractional ideal (2 + 46*j + 46*k, i + 46*j + 41*k, 128*j, 128*k)
```

```
Fractional ideal (2 + 46*j + 50*k, i + 50*j + 9*k, 64*j, 64*k)
Fractional ideal (2 + 46*j + 110*k, i + 110*j + 105*k, 128*j, 128*k)
Fractional ideal (2 + 62*j + 22*k, i + 22*j + 33*k, 64*j, 64*k)
Fractional ideal (2 + 62*j + 22*k, i + 22*j + 97*k, 128*j, 128*k)
Fractional ideal (2 + 94*j + 70*k, i + 70*j + 17*k, 128*j, 128*k)
Fractional ideal (2 + 110*j + 78*k, i + 78*j + 73*k, 128*j, 128*k)
Fractional ideal (2 + 110*j + 206*k, i + 206*j + 201*k, 256*j, 256*k)
Fractional ideal (2 + 174*j + 46*k, i + 46*j + 41*k, 256*j, 256*k)
sage: t2 = B.hecke_matrix(2); t2
33 x 33 dense matrix over Rational Field
sage: DiGraph(t2).plot()

sage: G1 = DiGraph(SupersingularModule(389).hecke_matrix(2))
sage: G2 = DiGraph(BrandtModule(389).hecke_matrix(2))
sage: G1.is_isomorphic(G2)
True
```

4.3 Computing Brandt Modules

The *Brandt module* $B(pM)$ of level pM for p a prime and M an integer not divisible by p is the free abelian group on the right ideal classes in an Eichler order of level M in the quaternion algebra ramified at p and ∞. This Brandt module is a module over the Hecke algebra, and is isomorphic to the group $X = \mathrm{Div}(X_0(Mp)^{\mathrm{ss}}_{\overline{\mathbb{F}}_p})$ of divisors on isomorphism classes of enhanced supersingular elliptic curves $\mathbf{E} = (E, C)$ over $\overline{\mathbb{F}}_p$, with C cyclic of order M.

The references I know of for how to *compute* $B(pM)$ are [**?**], the Magma source code, the Sage source code, and numerous papers on John Voight's website `http://www.cems.uvm.edu/~voight/`. The article [**?**] by David Kohel also has some useful theoretical background, but doesn't go into any real detail about how to actually compute $B(pM)$. Gonzalo Tornaría is also an excellent resource `http://www.cmat.edu.uy/~tornaria/`. For generalizations of Brandt modules to quaternion algebra over totally real number fields (with applications to computing Hilbert modular forms), see Lasina Dembele's publications page `http://www.uni-due.de/~hx0043/papers/paper.html`.

The rest of this section is an overview of the algorithms implemented in Sage. At the time of this writing, the Sage implementation of computation of Brandt modules over $\mathbb{Q}$ was significantly more efficient overall than the Magma's implementation.

4.3.1 Arithmetic

Let Q be the quaternion algebra with $i^2 = a$ and $j^2 = b$, where $a, b \in \mathbb{Q}$. We call a, b the *invariants* of Q.

In Sage, we represent an element $\theta \in Q$ by giving a 4-tuple of integers x, y, z, w and a demoninator d, all of MPIR C data type `mpz_t`, such that

$$\theta = \frac{1}{d}(x + yi + zj + wk)$$

We use the following formula for multiplication, which John Voight just sat down and scribbled on a piece of paper after thinking hard for a while at Sage Days 13.

Given two quaternion algebra elements

$$\theta = \frac{1}{d_1}(x_1 + y_1 i + z_1 j + w_1 k)$$

and

$$\nu = \frac{1}{d_2}(x_2 + y_2 i + z_2 j + w_2 k)$$

we compute their product as

$$\theta\nu = \frac{1}{d_3}(x_3 + y_3 i + z_3 j + w_3 k)$$

where $d_3 = d_1 d_2$ and

$$\begin{aligned} x3 &= t_1 + at_2 + b(t_3 - at_4) \\ y3 &= s_1(x_2 + y_2) - t_1 - t_2 + b(s_2(z_2 - w_2) - t_3 + t_4) \\ z3 &= t_5 - at_6 + t_7 + at_8 \\ w3 &= (x_2 - y_2)s_2 - t_5 + t_6 + s_1(z_2 + w_2) - t_7 - t_8 \end{aligned}$$

and where

$$\begin{aligned} t_1 &= x_1 x_2 \\ t_2 &= y_1 y_2 \\ t_3 &= z_1 z_2 \\ t_4 &= w_1 w_2 \\ t_5 &= x_2 z_1 \\ t_6 &= y_2 w_1 \\ t_7 &= x_1 z_2 \\ t_8 &= y_1 w_2 \\ s_1 &= x_1 + y_1 \\ s_2 &= z_1 + w_1 \end{aligned}$$

Ignoring denominators, this takes more integer addition operations but fewer integer multiplication operations (17) than the "straightforward" multiplication method (which takes 24 multiplies). There might be a way to optimize this formula further. The paper *The Complexity of the Quaternion Product*, 1975, Thomas Howell, proves that for $a = b = -1$, the product can be done in 8 multiplies and no less than 7. For us, in this special case, our formula reduces to 12 multiplies. Note that the prievously cited paper only addresses multiplying quaternions with $a = b = -1$. It would be interesting to see if there is a better algorithm in the general case (to do this, do a literature search and/or read the above paper and generalize the techniques).

To get a sense of speed, multiplying the following two quaternions with $a = -7, B = -11$ takes 1.3 *microseconds* on sage.math (a 2.6Ghz Xeon Dunnington).

```
sage: Q.<i,j,k> = QuaternionAlgebra(-7,-11)
sage: a = 9394 + 3939*i + 1293*j - 1933*k
sage: b = 39392 - 4928*i - 19394*j + 3912*k
sage: timeit('a*b')
625 loops, best of 3: 1.3 microseconds per loop
```

Using Karatsuba twice, Tom Boothby, Robert Bradshaw, and Craig Citro figured out how to multiply quaternions using only 16 multiplies, as follows. To multiply two quaternions r and s, we write $r = \alpha + \beta j$ and $s = \gamma + \delta j$ where $\alpha, \beta, \gamma, \delta \in K[i]$. Multiplication of elements of $K[i]$ can written as $(a + bi)(c +$

$di) = ac + (ac + bd)i + bdp$ where the middle term can be computed using one fewer multiplication via Karatsuba's trick $ad + bc = (a + b)(c + d) - ac - bd$. Now compute

$$(\alpha + \beta j)(\gamma + \delta j) = \alpha\gamma + \alpha\delta j + \beta j\gamma + \beta j\delta j = \alpha\gamma + (\alpha\delta + \beta\bar{\gamma})j + \beta\bar{\delta}q.$$

The middle term can be obtained in a similar manner as above:

$$\alpha\delta + \beta\bar{\gamma} = (\alpha + \bar{\beta})(\gamma + \delta) - \alpha\gamma - \overline{\beta\bar{\delta}} - (\bar{\beta}\gamma - \beta\bar{\gamma}).$$

The difference $\bar{\beta}\gamma - \beta\bar{\gamma}$ has "real part" zero (as conjugation yields its negative), so it can be computed with only two field multiplies.

In summary, writing $S = \alpha\gamma$, $T = \beta\bar{\delta}$, $U = (\alpha+\bar{\beta})(\gamma+\delta)$, and $R = \bar{\beta}\gamma - \beta\bar{\gamma}$ we have

$$(\alpha + \beta j)(\gamma + \delta j) = S + Tq + (U - S - \overline{T} - R)j.$$

Computing S, T, U takes three $K[i]$ multiplications, or 12 field multiplications, and the computation of R takes 2 more. The multiplication Tq takes 2 field multiplications, yielding a total of 16 multiplications in the basefield. A similar analysis shows that this formula uses a total of 27 field additions and subtractions, as well as a single doubling (to computing R).

4.3.2 Computing the Quaternion Algebra

Let A be the rational quaternion algebra ramified at p and ∞. Then we can take A to be $\mathbb{Q} < i, j, k >$ where $i^2 = a, j^2 = b$ and $ij = -ji = k$, and a, b are determined as follows:

$$(a, b) = \begin{cases} (-1, -1) & \text{if } p = 2 \\ (-1, -p) & \text{if } p \equiv 3 \pmod 4 \\ (-2, -p) & \text{if } p \equiv 5 \pmod 8 \\ (-p, -q) & \text{if } p \equiv 1 \pmod 8, \end{cases}$$

where in the last case $q \geq 3$ is the smallest prime with $q \equiv 3 \pmod 4$ and $\left(\frac{p}{q}\right) = -1$. See [**?**, Prop. 5.1] for references about how to prove this using Hilbert symbols.

Example 4.11. We compute quaternion algebras of each of the above types.

```
sage: BrandtModule(2).quaternion_algebra()
Quaternion Algebra (-1, -1) with base ring Rational Field
sage: BrandtModule(3).quaternion_algebra()
Quaternion Algebra (-1, -3) with base ring Rational Field
sage: BrandtModule(5).quaternion_algebra()
Quaternion Algebra (-2, -5) with base ring Rational Field
sage: BrandtModule(17).quaternion_algebra()
Quaternion Algebra (-17, -3) with base ring Rational Field
```

4.3.3 Computing a Maximal Order

Let A with $i^2 = a$, $j^2 = b$ be the rational quaterion algebra ramified at p and ∞ from Section **??**. A maximal order R for A has basis

$$(b_1, b_2, b_3, b_4) = \begin{cases} (1+i+j+k)/2, i, j, k & \text{if } p = 2, \\ (1+j)/2, (i+k)/2, j, k & \text{if } p \equiv 3 \pmod 4, \\ (1+j+k)/2, (i+2j+k)/4, j, k & \text{if } p \equiv 5 \pmod 8, \\ (1+j)/2, (i+k)/2, (j+zk)/b, k & \text{if } p \equiv 1 \pmod 8, \end{cases}$$

where in the last case z is any integer such that $b \mid (z^2 p + 1)$.

To prove that the above are maximal orders, one tediously checks that the lattice they span has discriminant p and is a subring that contains 1. The discriminant of R is just the discriminant of the reduced trace pairing on a basis for R. The conjugate of $x + iy + zj + wk$ is $x - iy - zj - wk$. The reduced trace is $\mathrm{Tr}(x + iy + zj + wk) = 2x$, and the reduced trace pairing is $\langle c, d \rangle = \mathrm{Tr}(c\overline{d})$.

Example 4.12. We compute a maximal order in each of the above cases:

```
sage: BrandtModule(2).maximal_order().basis()
(1/2 + 1/2*i + 1/2*j + 1/2*k, i, j, k)
sage: BrandtModule(3).maximal_order().basis()
(1/2 + 1/2*j, 1/2*i + 1/2*k, j, k)
sage: BrandtModule(5).maximal_order().basis()
(1/2 + 1/2*j + 1/2*k, 1/4*i + 1/2*j + 1/4*k, j, k)
sage: BrandtModule(17).maximal_order().basis()
(1/2 + 1/2*j, 1/2*i + 1/2*k, -1/3*j - 1/3*k, k)
```

We use the above algorithm to compute a maximal order for $p = 7$.

```
sage: p = 7
sage: Q.<i,j,k> = QuaternionAlgebra(-1,-p); Q
Quaternion Algebra (-1, -7) with base ring Rational Field
sage: Q.discriminant()
7
sage: R = Q.quaternion_order([(1+j)/2, (i+k)/2, j, k]); R
Order of Quaternion Algebra (-1, -7) with base ring
  Rational Field with basis (1/2 + 1/2*j, 1/2*i + 1/2*k, j, k)
sage: R.discriminant()
7
sage: Q.quaternion_order([1,i,j,k]).discriminant()
28
```

4.3.4 Computing an Order of Level $p^{2r+1}M$

Let A be the quaternion algebra ramified at p, ∞ with $i^2 = a$, $j^2 = b$.

Let M be an integer coprime to p and let $r \geq 0$ be an integer. The following definition from Pizer's paper [**?**] is more general than the definition of Eichler order of level M given above; note also that p is included in the "level" below.

Definition 4.13 (Level of Order)**.** An order R has level $N = p^{2r+1}M$ if for all primes $q \neq p$ there is an element $z \in A \otimes \mathbb{Z}_q$ such that

$$zRz^{-1} = \begin{pmatrix} \mathbb{Z}_q & \mathbb{Z}_q \\ N\mathbb{Z}_q & \mathbb{Z}_q \end{pmatrix},$$

and there is $z \in A \otimes \mathbb{Z}_p$ such that

$$zRz^{-1} = \left\{ \begin{pmatrix} \alpha & p^r\beta \\ p^{r+1}\beta^\sigma & \alpha^\sigma \end{pmatrix} : \alpha, \beta \in \mathbb{Z}_{p^2} \right\},$$

where $\mathbb{Z}_{p^2}$ is the ring of integers in the unique unramified quadratic extension of $\mathbb{Q}_p$ and σ is its nontrivial automorphism.

To compute an order of level pM, we proceed a prime at a time. First, let R be the maximal order in A constructed in Section **??** above. If I is a left R-ideal, the right order S associated to I is the set of elements $x \in R$ such that $Ix \subset I$. Thus S is a subring of R and I is a right S-ideal (note that $I \subset S$ because if $x \in I$ then $Ix \subset I$, since $I \subset R$ and I is a left ideal).

Let q be a prime divisor of M, and let $t = \mathrm{ord}_q(M)$. Find an element $y \in R$ such that

$$f = X^2 - \mathrm{Tr}(y)X + \mathrm{Norm}(y) \in \mathbb{F}_q[X]$$

has distinct roots in $\mathbb{F}_q$, where Tr is the reduced trace and Norm is the reduced norm. We can find x either by trying random elements, or systematically running through linear combinations of the basis for R with coefficients between 0 and $q-1$. Once we find such an element y, let and let $a \in \mathbb{F}_q$ be one of the roots of f above. Let I be the left R-ideal generated by q^t and $(x-a)^t$. Let S be the right order in R associated to I. Then $S \otimes \mathbb{Z}_\ell = R \otimes \mathbb{Z}_\ell$ for primes $\ell \neq q$ and $S \otimes \mathbb{Z}_q \subset R \otimes \mathbb{Z}_q$. [[finish giving an argument that this works! – no known one in literature, but should be obvious]] Finally, we replace R by S.

To compute an order of level $p^{2r+1}M$ we proceed as above to obtain an order of level pM. [[I haven't worked out the details yet, but one could always just iterate through all sublattices of index $p^?$ and for each check of ...]]

Example 4.14. We compute an order of level $N = pM$ in several cases. Note that Sage does not currently (June 2009) have an algorithm in case $r > 0$.

```
sage: BrandtModule(2,7).order_of_level_N().basis()
(1/2 + 1/2*i + 1/2*j + 1/2*k, i + 5*k, j + 3*k, 7*k)
sage: BrandtModule(3,7).order_of_level_N().basis()
(1/2 + 1/2*j, 1/2*i + 3/2*k, j, 7*k)
sage: BrandtModule(5,7).order_of_level_N().basis()
(1/2 + 1/2*j + 9/2*k, 1/4*i + 1/2*j + 17/4*k, j + 2*k, 7*k)
sage: BrandtModule(17,7).order_of_level_N().basis()
(1/2 + 1/6*j + 11/3*k, 1/2*i + 13/2*k, 1/3*j + 1/3*k, 7*k)
```

4.3.5 Equivalence of Right Ideal

Let R be an order of level $N = p^{2r+1}M$ in the quaternion algebra A ramified at p and ∞. If I and J are nonzero right R-ideals, then we say that I is equivalent to J, written $I \sim J$ if there is $a \in A$ such that $I = aJ$. The Brandt module $B(p^{2r+1}, M)$ is the free abelian group on the set of right ideal classes of R.

Let I be a right ideal of R with fixed choice of basis b_1, b_2, b_3, b_4. The *Gram matrix* G of I is the matrix whose i, j entry is $2\,\mathrm{Tr}(b_i b_j)$, where Tr is the reduced trace. Let G' be obtained from G by rescaling so that all entries are integers and the gcd of all entries is 1. The normalized θ-series associated to a right ideal I of R is a formal power series $\theta_I \in \mathbb{Z}[[q]]$. It is the θ series associated to the quadratic form with Gram matrix G', where the θ series of a quadratic form has the property that the coefficient of q^n is the number of vectors of length n in the abstract lattice with that inner product matrix.

Given two right R-ideals I and J, we determine whether or not they are equivalent as follows. First, we compute the normalized θ series θ_I and θ_J associated to I, J to some precision. If they are not equal, then definitely $I \not\sim J$. If they are equal, then I might or might not be equivalent to J (theta are inequivalent ideals with equal θ series). In that case, we compute the product $I\overline{J}$, then use [**?**, Prop. 1.18] that $I \sim J$ if and only if there is $\alpha \in I\overline{J}$ with $\mathrm{Norm}(\alpha) = \mathrm{Norm}(I)\,\mathrm{Norm}(J)$. We can decide the latter by computing the coefficient of q in the normalized theta series associated to $I\overline{J}$.

4.3.6 The Action of Hecke Operators

As above, R is an order of level $N = p^{2r+1}M$ in the quaternion algebra A ramified at p and ∞. For right ideals I, J of R, say a nonzero group homomorphism $\varphi : I \to J$ is *cyclic of degree* n if

$$J/\varphi(I) \cong \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}.$$

For any integer n with $\gcd(n, pM) = 1$, we have

$$T_n([I]) = \sum_{\varphi} [J],$$

where the sum is over the cyclic R-module homomorphisms $\varphi : I \to J$ (up to isomorphisms in J). We can embed I, J into A as fractional right R-ideals such that the homomorphism φ is an inclusion

$$I \hookrightarrow J \hookrightarrow n^{-1}I,$$

and we can instead compute the Hecke operator T_n by summing over cyclic supermodules $I \hookrightarrow J$.

Efficiently enumerating the cyclic supermodules J containing I is tricky (it took me a number of hours to come up with a fast algorithm). We assume that n is prime. The basic idea is to reduce everything to linear algebra modulo n, work in the module I/nI, which is also a 4-dimensional $\mathbb{F}_n$-vector space. We explicitly compute the action of generators of R on I/nI in terms of matrices mod n, and use this structure to explicitly write down all cyclic submodules.

4.3.7 Computing all Right Ideal Classes

To enumerate all right ideal classes, we proceed as follows. First we let I be the unit ideal. Let ℓ be the smallest prime that doesn't divide $p^{2r+1}M$. Using the algorithm of Section **??**, compute the ideals J appearing in the sum $T_\ell([I]) = \sum_\varphi [J]$. For each ideal, we check whether it is equivalent to any ideal seen so far, using the algorithm of Section **??**. If not, we add it to our list of right ideal class representatives. We continue applying T_ℓ to new right ideal representatives until we don't see any new ones. At this point, we must have enumerated all right ideal classes, since a theorem of Serre implies that the graph associated to the Hecke operator T_ℓ is connected [[the proof uses that the Hecke graph is regular and using an Eisenstein series one sees something relevant about an eigenvalue of that matrix]].

Example 4.15. We explicitly compute the two distinct right ideal classes in an order of level 14 in the quaternion algebra ramified at 2 and ∞ using the algorithm described above.

```
sage: B = BrandtModule(2,7)
sage: R = B.order_of_level_N()
sage: I = R.unit_ideal()
sage: M = B.cyclic_supermodules(I,3); M
[Fractional ideal (1/2 + 1/2*i + 3/2*j + 7/2*k, i + 2*j + 11*k, 3*j + 9*k, 21*k),
 Fractional ideal (1/2 + 1/2*i + 1/2*j + 15/2*k, i + 2*j + 4*k, 3*j + 9*k, 21*k),
 Fractional ideal (1/2 + 1/2*i + 3/2*j + 35/2*k, i + j + k, 3*j + 9*k, 21*k),
 Fractional ideal (1/2 + 1/2*i + 5/2*j + 27/2*k, i + j + 8*k, 3*j + 9*k, 21*k)]
```

The elements of M are the ideals J with $I \subset J$ cyclic of order 3. One of them is equivalent to the unit ideal and the others aren't:

```
sage: [A.is_equivalent(I) for A in M]
[False, False, True, False]
```

Computing theta series suggests that all the ideals not equivalent to the unit ideal are equivalent to each other.

```
sage: for A in M: print A.theta_series(8)
1 + 6*q^2 + 18*q^3 + 18*q^5 + 6*q^6 + 42*q^7 + O(q^8)
1 + 6*q^2 + 18*q^3 + 18*q^5 + 6*q^6 + 42*q^7 + O(q^8)
1 + 6*q + 6*q^3 + 6*q^4 + 18*q^5 + 18*q^6 + 48*q^7 + O(q^8)
1 + 6*q^2 + 18*q^3 + 18*q^5 + 6*q^6 + 42*q^7 + O(q^8)
```

And indeed they are.

```
sage: [A.is_equivalent(M[0]) for A in M]
[True, True, False, True]
```

We also apply T_3 to the non-unit ideal class, and find that this gives us nothing new.

```
sage: [S.is_equivalent(M[0]) or S.is_equivalent(I) for S in B.cyclic_supermodules(M[0],3)]
[True, True, True, True]
```

References

[AB04] Montserrat Alsina and Pilar Bayer, *Quaternion orders, quadratic forms, and Shimura curves*, CRM Monograph Series, vol. 22, American Mathematical Society, Providence, RI, 2004. MR MR2038122 (2005k:11226)

[Cre97] J. E. Cremona, *Algorithms for modular elliptic curves*, second ed., Cambridge University Press, Cambridge, 1997, `http://www.maths.nott.ac.uk/personal/jec/book/`.

[DS05] Fred Diamond and Jerry Shurman, *A first course in modular forms*, Graduate Texts in Mathematics, vol. 228, Springer-Verlag, New York, 2005.

[Fal86] G. Faltings, *Finiteness theorems for abelian varieties over number fields*, Arithmetic geometry (Storrs, Conn., 1984), Springer, New York, 1986, Translated from the German original [Invent. Math. **73** (1983), no. 3, 349–366; ibid. **75** (1984), no. 2, 381; MR 85g:11026ab] by Edward Shipz, pp. 9–27. MR 861 971

[Koh] D. R. Kohel, *Hecke module structure of quaternions*, In K. Miyake, ed., *Class Field Theory – Its Centenary and Prospect*, The Advanced Studies in Pure Mathematics Series, Math Soc. Japan.

[Man72] J. I. Manin, *Parabolic points and zeta functions of modular curves*, Izv. Akad. Nauk SSSR Ser. Mat. **36** (1972), 19–66. MR 47 #3396

[Mer94] L. Merel, *Universal Fourier expansions of modular forms*, On Artin's conjecture for odd 2-dimensional representations, Springer, 1994, pp. 59–94.

[Mil86] J. S. Milne, *Abelian varieties*, Arithmetic geometry (Storrs, Conn., 1984), Springer, New York, 1986, pp. 103–150.

[MTT86] B. Mazur, J. Tate, and J. Teitelbaum, *On p-adic analogues of the conjectures of Birch and Swinnerton-Dyer*, Invent. Math. **84** (1986), no. 1, 1–48. MR MR830037 (87e:11076)

[Piz80] A. Pizer, *An algorithm for computing modular forms on* $\Gamma_0(N)$, J. Algebra **64** (1980), no. 2, 340–390.

[Rib80] K. A. Ribet, *Twists of modular forms and endomorphisms of abelian varieties*, Math. Ann. **253** (1980), no. 1, 43–62. MR 82e:10043

[Rib92] ———, *Abelian varieties over* $\mathbf{Q}$ *and modular forms*, Algebra and topology 1992 (Taejŏn), Korea Adv. Inst. Sci. Tech., Taejŏn, 1992, pp. 53–79. MR 94g:11042

[Ser73] J-P. Serre, *A Course in Arithmetic*, Springer-Verlag, New York, 1973, Translated from the French, Graduate Texts in Mathematics, No. 7.

[Shi73] G. Shimura, *On the factors of the jacobian variety of a modular function field*, J. Math. Soc. Japan **25** (1973), no. 3, 523–544.

[Shi94] ———, *Introduction to the arithmetic theory of automorphic functions*, Princeton University Press, Princeton, NJ, 1994, Reprint of the 1971 original, Kan Memorial Lectures, 1.

[Vig80] Marie-France Vignéras, *Arithmétique des algèbres de quaternions*, Lecture Notes in Mathematics, vol. 800, Springer, Berlin, 1980. MR MR580949 (82i:12016)