A database of modular forms on noncongruence subgroups

David Berghaus, Hartmut Monien, Danylo Radchenko

A DATABASE OF MODULAR FORMS ON NONCONGRUENCE SUBGROUPS

DAVID BERGHAUS, HARTMUT MONIEN, AND DANYLO RADCHENKO

Abstract. We present a database of several hundred modular forms up to and including weight six on noncongruence subgroups of index ≤ 17. In addition, our database contains expressions for the Belyi map for genus zero subgroups and equations of the corresponding elliptic curves for genus one subgroups and numerical approximations of noncongruence Eisenstein series to 1500 digits precision.

1. Introduction

The theory of modular forms on noncongruence subgroups is still not well understood (for recent progress see [6, 7]). In this work we provide a large number of computed examples in the hope that these will lead to new observations and conjectures. Building a database of modular forms on noncongruence subgroups has been listed as one of the goals for future research in the AIM Workshop *Noncongruence modular forms and modularity* [17]. The usefulness of computer generated data to investigate modular forms on noncongruence subgroups has already been demonstrated by Atkin and Swinnerton-Dyer [1] who formulated key conjectures based on just a handful of computed examples. Despite this, very few modular forms on noncongruence subgroups have been computed. The known examples were typically restricted to noncongruences character groups [12, 15] or to noncongruence subgroups of very low index [13] and a few large index genus zero cases [19].

In this work we apply the algorithm of [3] to make a database of modular forms on noncongruence subgroups. This complements the existing related databases such as the database of classical modular forms on congruence subgroups [5], the database of Hilbert modular forms [11], and the database of Belyi maps [20].

2. Background and Notation

2.1. **Modular Forms.** Let $\mathrm{SL}(2,\mathbb{Z})$ be the group of 2×2 integer matrices with determinant 1 and let G be a finite index subgroup of $\mathrm{SL}(2,\mathbb{Z})$. A modular form of weight k for the group G is a holomorphic function on the upper half plane $\mathcal{H} := \{\tau \in \mathbb{C} \,|\, \mathrm{Im}(\tau) > 0\}$ that satisfies the transformation law

$$f(\gamma(\tau)) = (c\tau + d)^k f(\tau)\,, \qquad \text{for all } \gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in G\,, \tag{2.1}$$

where

$$\gamma(\tau) := \frac{a\tau + b}{c\tau + d}\,, \tag{2.2}$$

and such that $(c\tau+d)^{-k} f(\gamma(\tau))$ is bounded as $\mathrm{Im}(\tau) \to \infty$ for all $\gamma \in \mathrm{SL}(2,\mathbb{Z})$. In this paper we will restrict to the case when there is no multiplier system and when k is an even integer and without loss of generality we may instead pass to the quotient $\mathrm{PSL}(2,\mathbb{Z}) := \mathrm{SL}(2,\mathbb{Z})/\{\pm 1\}$, which we denote by Γ and we consider G as a finite index subgroup of Γ. For more insight into the extensive theory of classical modular forms we refer to the books [8] by Cohen and Strömberg and [10] by Diamond and Shurman, as well as the lecture notes [26] by Zagier. Our setup and notation follows the one used in [3].

The subgroup $\Gamma(N) \leqslant \Gamma$, defined by

$$\Gamma(N) := \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \equiv \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \pmod{N} \quad \text{and} \quad \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma \right\}, \tag{2.3}$$

is called the *principal congruence subgroup of level N*. A finite index subgroup $G \leqslant \Gamma$ is called *noncongruence* if it does not contain $\Gamma(N)$ for any $N \geq 1$.

Let G be any finite index subgroup of Γ and let f be a modular form of weight $k \in 2\mathbb{Z}_{>0}$. Then f has a Fourier expansion

$$f(\tau) = \sum_{n=0}^{\infty} a_n q_h^n \,, \tag{2.4}$$

where $q_h := \exp(2\pi i\tau/h)$ and h denotes the width of the cusp at ∞. One can choose bases for spaces of modular forms on noncongruence subgroups with Fourier coefficients defined over $\overline{\mathbb{Q}}$, and moreover their coefficients can be written in the form (see [1, 21])

$$a_n = u^n b_n \,, \tag{2.5}$$

with b_n and u^h belonging to a certain number field K. We fix a choice v of a generator for K over $\mathbb{Q}$ (i.e., $K = \mathbb{Q}(v)$ with v an algebraic number). We also denote the number field that contains all a_n by $L = \mathbb{Q}(w)$.

2.2. **Belyi Maps.**

Definition 2.1 (Belyi Map)**.** Let X be a compact Riemann surface. A holomorphic function

$$f \,:\, X \to \mathbb{P}^1(\mathbb{C}) \,, \tag{2.6}$$

is called a *Belyi map* if it is unramified away from three points.

The covering map

$$R \,:\, X(G) \to X(\Gamma) \overset{j}{\cong} \mathbb{P}^1(\mathbb{C}) \,, \tag{2.7}$$

is a Belyi map, where $X(G) = G\backslash\overline{\mathcal{H}}$ is the modular curve of G. If G is a genus zero subgroup then the covering map $R(j_G)$ is a rational function in the Hauptmodul j_G that branches over the images of the elliptic points and the cusps. Note that the Belyi map can be defined over the same field K as above.

2.3. **Signatures and Passports.**

Definition 2.2 (Signature)**.** We define the *signature* of $G \leqslant \Gamma$ to be the tuple $(\mu, g, n(c), n(e_2), n(e_3))$, where μ denotes the index of G in Γ, g denotes the genus of the fundamental domain $\mathcal{F}(G)$, $n(c)$ denotes the number of cusps of G and $n(e_2)$ and $n(e_3)$ denote the number of elliptic points of order two and three, respectively.

The signature can be used to distinguish between different *types* of subgroups. A further distinction can be made using the notion of *passport*:

Definition 2.3 (Passport)**.** Given a set of subgroups of equal signature that, by Millington's theorem [18], correspond to tuples $(\sigma_S, \sigma_R, \sigma_T)$ which are representatives of S_μ conjugacy classes, we say that two subgroups belong to the same passport if their *monodromy groups* (i.e., the permutation group generated by σ_S and σ_R) are conjugate in S_μ.

Passports are useful to know because the action of the absolute Galois group on Belyi maps preserves the passport and hence the size of the passport (i.e., the number of elements inside a passport) provides an upper bound on the degree of the number field K.

3. Numerical Computation of Elliptic Curves

In [3] we showed how Fourier expansions of modular forms on noncongruence subgroups can be computed numerically to high precision. In this section we show how one can compute from this an equation for the elliptic curve corresponding to a genus one noncongruence subgroup. We use an approach similar to that described by Cremona [9, Chapter II]. Let $f \in S_2(G)$ denote a nontrivial weight two cusp form for a genus one group G (note that for genus one groups $\dim(S_2) = 1$ which means that f is unique up to normalization). The Fourier expansion of f at all cusps can be computed numerically by applying the algorithm of [3]. We then define the integral

$$I_f(\alpha, \beta) := \int_\alpha^\beta 2\pi i f(\tau) d\tau \,, \tag{3.1}$$

where $\alpha, \beta \in \overline{\mathcal{H}}$ and

$$I_f(\tau_0) := I_f(\tau_0, i\infty) = \int_{\tau_0}^{i\infty} 2\pi i f(\tau) d\tau = -\sum_{n=1} \frac{a_n}{n} \exp(2\pi i n \tau_0/h) \,, \tag{3.2}$$

where h denotes the cusp width. Then the period map P_f, defined by

$$P_f(\gamma) := I_f(\tau_0, \gamma(\tau_0)) = I_f(\tau_0) - I_f(\gamma(\tau_0)) \,, \qquad \gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in G \,, \tag{3.3}$$

is independent of the choice of the base point τ_0 and defines a homomorphism from G to the period lattice $\Lambda_f = \mathbb{Z}w_1 + \mathbb{Z}w_2$, with $w_1, w_2 \in \mathbb{C}$ (see [9]).

3.1. **Evaluation of $P_f(\gamma)$.** Since we work numerically, the base point τ_0 should be chosen in a way that maximizes precision when evaluating $P_f(\gamma)$. This is achieved by maximizing the minimum among the imaginary parts of τ_0 and $\gamma(\tau_0)$. If G has a single cusp, we choose

$$\tau_0 = -\frac{d}{c} + \frac{1}{c} i \in \mathcal{H} \,, \tag{3.4}$$

for $c > 0$ (if $c = 0$ then $P_f(\gamma) = 0$ and if $c < 0$ then we simply choose the other representative $-\gamma$ of γ in $\mathrm{PSL}(2, \mathbb{Z})$). Note that for this choice of τ_0 we have $\gamma(\tau_0) = a/c + i/c$ which means that the imaginary parts of τ_0 and $\gamma(\tau_0)$ are equal.

If G has multiple cusps, we denote by A_p the cusp normalizer associated to cusp p (using the convention that $A_p(i\infty) = p$) and denote cusp expansions at p by $f_p = f|_2 A_p$. Then we obtain

$$P_f(\gamma) = I_{f_p}(\tau_{0,p}) - I_{f_p}((A_p^{-1}\gamma A_p)(\tau_{0,p})) \,. \tag{3.5}$$

Let $\gamma_p := A_p^{-1}\gamma A_p = \begin{pmatrix} a_p & b_p \\ c_p & d_p \end{pmatrix}$. Then we choose

$$\tau_{0,p} = -\frac{d_p}{c_p} + \frac{1}{c_p} i \in \mathcal{H} \,, \tag{3.6}$$

and evaluate $P_f(\gamma)$ using the expansion from the cusp p for which the value $c_p \cdot h_p$ is minimized (where h_p denotes the cusp width of cusp p).

3.2. **Computation of Λ_f.** The approach for computing the period lattice Λ_f numerically can be summarized as follows:

1. Compute the set $S = \{P_f(\gamma_1), P_f(\gamma_2), ...\}$ where γ_j denote a set of generators for G.
2. Remove (approximate) zeros (which are trivially on the lattice) and (approximate) duplicate elements from the set S.
3. Choose any two linearly independent elements $w_1, w_2 \in S$ as a first guess for the basis of Λ_f.

(4) Express the remaining elements as elements in $\mathbb{Q}[w_1, w_2]$ by solving

$$\begin{pmatrix} \mathrm{Re}(w_1) & \mathrm{Re}(w_2) \\ \mathrm{Im}(w_1) & \mathrm{Im}(w_2) \end{pmatrix} \cdot \begin{pmatrix} r_j \\ s_j \end{pmatrix} = \begin{pmatrix} \mathrm{Re}(P_f(\gamma_j)) \\ \mathrm{Im}(P_f(\gamma_j)) \end{pmatrix}, \tag{3.7}$$

to obtain $P_f(\gamma_j) = r_j w_1 + s_j w_2$ with $r_j, s_j \in \mathbb{Q}$.

(5) Find the least common denominator λ so that $\lambda r_j, \lambda s_j \in \mathbb{Z}$ for all j.

(6) Find a basis matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ for the $\mathbb{Z}^2$-submodule spanned by $\lambda \cdot (r_j, s_j)$ by computing the Hermite normal form.

(7) Switch to the lattice basis

$$w_1, w_2 \mapsto \frac{aw_1 + bw_2}{\lambda}, \frac{cw_1 + dw_2}{\lambda}. \tag{3.8}$$

(8) Compute $\tau := w_1/w_2$ with $\mathrm{Im}(\tau) > 0$ (interchanging w_1 and w_2 if necessary) and transform τ to the fundamental domain of Γ to obtain a lattice $\mathbb{Z} + \mathbb{Z}\tau$ equivalent to Λ_f.

Remark 3.1. The precision of the elements $S = \{P_f(\gamma_1), P_f(\gamma_2), ...\}$ depends on the choice of the generator representatives (which we compute using Sage). We are unaware of any algorithm that produces generators that are optimal in the sense that τ_0 has maximal possible imaginary parts, see Section 3.1. For this reason we conjugate G by $\sigma = (1\,j)$, for $j = 2, ..., \mu$ (in terms of the permutation group representation of G) and choose the representative for which the height of τ_0 becomes maximal. While this choice might not be ideal, because it may happen that τ_0 is below the height of the fundamental domain, given by $\mathrm{Im}(\mathcal{F}(G)) = \sqrt{3}/(2h_{\max})$ where $h_{\max}$ denotes the largest cusp width (which means that we cannot evaluate $P_f(\gamma)$ to full precision), in practice the precision has always been sufficient to identify the elliptic curve.

3.3. **Computation of the elliptic curve.** Once the period lattice Λ_f has been obtained the computation of the elliptic curve becomes straightforward. We first compute the elliptic invariants

$$g_2(\tau) = 60G_4(\tau) = 60 \sum_{(m,n)\neq(0,0)} (m + n\tau)^{-4}, \tag{3.9}$$

where $G_4(\tau) = (\pi^4/45)E_4(\tau)$ and

$$g_3(\tau) = 140G_6(\tau) = 140 \sum_{(m,n)\neq(0,0)} (m + n\tau)^{-6}, \tag{3.10}$$

where $G_6(\tau) = (2\pi^6/945)E_6(\tau)$ using Arb [14]. Then the j-invariant is given by

$$j(\tau) = 1728 \frac{g_2^3}{g_2^3 - 27g_3^2}. \tag{3.11}$$

After that we identify $j(\tau)$ as an algebraic number in K (using the LLL algorithm [16]) and use Sage [22] to obtain an equation for the elliptic curve.

Example 3.2. Consider the noncongruence subgroup G with monodromy group $\mathrm{PSL}_2(\mathbb{F}_8)$ and signature $(9, 1, 1, 1, 0)$ that corresponds to the admissible permutation pair (σ_S, σ_R) with $\sigma_S = (1)(2\,5)(3\,7)(4\,8)(6\,9)$ and $\sigma_R = (1\,2\,6)(3\,8\,5)(4\,9\,7)$. We find that $\tau = 0.332234... + 0.744371...\,i$ and the value of the j-invariant is $\approx -1159088625/2097152$, from which we get the elliptic curve

$$E\colon y^2 + xy + y = x^3 - x^2 - 95x - 697 \tag{3.12}$$

of conductor 162. We can corroborate this result by computing the three weight 6 cusp forms of $S_6(G)$ (which we denote by x, y, z here) and observing that they satisfy the relation

$$-xy^2 + (-2w)y^3 + x^2z + (-w)xyz + (19w^2)y^2z + (-3w^3)yz^2 + (15w^4)z^3 = 0\,, \tag{3.13}$$

for $w = (-27/256)^{1/9}$. Substituting $x \mapsto xw$, $z \mapsto z/w$ and dividing by w we obtain the cubic

$$-xy^2 - 2y^3 + x^2z - xyz + 19y^2z - 3yz^2 + 15z^3 = 0\,, \tag{3.14}$$

defined over $\mathbb{Q}$ from which we obtain the same equation for E using Sage.

4. Database Structure

A database of noncongruence subgroups of index $\mu \leq 17$ has been computed by Strömberg [24] and can be obtained at [23]. We have used it as a foundation for our database.

Remark 4.1. We have developed an algorithm that can list subgroups for indices beyond those computed in [23, 24]. With its help we were able to generate all subgroups of index $\mu \leq 30$. However later we discovered a paper by Vidal [25] which describes a much more efficient and essentially optimal algorithm for the same task. We plan to extend the database with higher index subgroups in the foreseeable future.

4.1. **Database labels.** Each database entry is labeled by the signature of the elements contained in the passport, followed by the id of the passport, as well as a letter that labels the Galois orbit.

Example 4.2. Consider the passport of signature $(15, 1, 2, 1, 0)$ and monodromy group S_{15} which contains three elements (this is the fourth passport of this signature in the database [23] so we give it the id 3). This passport decomposes into two Galois orbits: The first one with label *15_1_2_1_0_3_a* for which $K = \mathbb{Q}(v)$ where $v^2 - v - 1 = 0$ and the second one with label *15_1_2_1_0_3_b* for which $K = \mathbb{Q}$.

4.2. **Permutation triple normalization.** Each database entry is unique up to a normalization of the permutation triple (i.e., conjugation in $\mathrm{SL}_2(\mathbb{Z})$). We usually normalized σ_T in a way that the cycle type is sorted in decreasing order and that the labels are sorted in increasing order. This means that the largest cusp is placed at infinity. While this causes the number field L to be of larger degree (and hence the arithmetic to be slower), the factored expressions in u and v typically involve smaller factors and are hence are more easily readible.

Example 4.3. Consider the permutation $(1\,2\,3\,4)(5\,6\,7)(8\,9)$ which has a cycle type of $(4, 3, 2)$ that is sorted in decreasing order.

There are three exceptions to the above strategy. The first is given for the case when the subgroup G has multiple cusps of equal width. Placing one of these cusps at infinity may cause K to be of a larger degree, due to the broken symmetry. In this case we therefore normalize σ_T so that the largest cusp with unique cusp width is placed at infinity. The second exception occurs if G has a cusp of (unique) width 1. In this case we place the cusp of width 1 at infinity because it is convenient that $K = L$. The third exception occurs for genus one subgroups where we do not sort the labels of σ_T in increasing order in case another normalization leads to better precision when computing the elliptic curve (see Section 3).

4.3. **The data that has been computed for each database element.** For each database element we have computed the curve (either the Belyi map for genus zero subgroups or the elliptic curve for genus one subgroups), as well as the Fourier expansions for the Hauptmoduls, cusp forms, and modular forms up to and including weight 6. For genus zero subgroups we have achieved this by applying the algorithms of [3, Section 5] over a number

field L. For higher genera subgroups we have computed numerical approximations of the corresponding Fourier expansions to 1500 digits of precision using the method of [3, Section 4.3.2] and from this guessed the closed-form solutions using the LLL algorithm. We remark that for reasons of efficiency, it is not required to compute all forms *from scratch*. Instead, we compute products between cusp forms and modular forms to generate (some of the) higher weight forms. Additionally, we have computed numerical approximations of the basis factors that map M_k onto E_k to 1500 digits precision using an algorithm described in [4]. Interestingly, we were unable to find any algebraic dependencies among these expressions.

4.4. **A complete example.** For an explicit example, let us take a look at the database entry 15_1_2_1_0_2_a.

- G: The subgroup G corresponds to the permutation group generated by
$$\sigma_S = (1\,15)(2\,12)(3\,7)(4\,9)(5\,13)(6\,10)(8\,14)(11)\,,$$
$$\sigma_R = (1\,11\,12)(2\,13\,6)(3\,8\,15)(4\,10\,7)(5\,14\,9)\,,$$
$$\sigma_T = (1\,3\,4\,5\,6\,7\,8\,9\,10\,2)(11\,12\,13\,14\,15)\,.$$
- Monodromy Group: $(C_3 \times C_3 \times C_3 \times C_3) : (C_2 \times A_5)$.
- K: $v = 1$ (i.e., $K = \mathbb{Q}$).
- Embeddings: $v = 1$.
- u: $(-3125/14348907)^{1/10}$ with an embedding $-0.409269... - 0.132979... \cdot i$.
- L: $w^8 - 5/27w^6 + 25/729w^4 - 125/19683w^2 + 625/531441$ with $w = -0.409269... - 0.132979... \cdot i$.
- Curve: $y^2 + xy + y = x^3 - x^2 + 7x - 103$.
- Fourier expansions (up to ~ 700 terms):
  - M_2:
    - $m_0 = 1 + 48/5u^2q_{10}^2 - 528/5u^3q_{10}^3 - 2448/25u^4q_{10}^4 - 3096/25u^5q_{10}^5 + ...$
    - $m_1 = q_{10} + uq_{10}^2 - 21/5u^2q_{10}^3 + 101/5u^3q_{10}^4 + 816/25u^4q_{10}^5 + ...$
  - S_2:
    - $s_0 = q_{10} + uq_{10}^2 - 21/5u^2q_{10}^3 + 101/5u^3q_{10}^4 + 816/25u^4q_{10}^5 + ...$ (this is the same as m_1 in this case).
  - E_2:
    - e_0: We give the Eisenstein basis matrix $[1, 0.356085... + 0.115699... \cdot i]$ (up to 1500 terms), meaning that $e_0 = m_0 + (0.356085... + 0.115699... \cdot i)m_1$. Note that this basis is already in canonical normalization.
  - M_4:
    - $m_0 = 1 - 688747536/625u^{10}q_{10}^{10} + ...$ (this is an oldform from Γ, namely the weight 4 Eisenstein series for $\mathrm{SL}(2, \mathbb{Z})$).
    - $m_1 = q_{10} - 70111/110u^4q_{10}^5 + ...$
    - $m_2 = q_{10}^2 - 621/22u^3q_{10}^5 + ...$
    - $m_3 = q_{10}^3 + 237/22u^2q_{10}^5 + ...$
    - $m_4 = q_{10}^4 - 115/22uq_{10}^5 + ...$
  - S_4:
    - $s_0 = q_{10} - 65u^3q_{10}^4 - 1488/5u^4q_{10}^5 + ...$
    - $s_1 = q_{10}^2 - 27/5u^2q_{10}^4 + ...$
    - $s_2 = q_{10}^3 - uq_{10}^4 + 16u^2q_{10}^5 + ...$
  - E_4:
    - $e_0 = m_0$.
    - $e_1 = m_1 + (12.068314... + 3.921233... \cdot i)m_2 + (106.205344... + 77.162699... \cdot i)m_3 + (-61.318023... - 84.397019... \cdot i)m_4$.
    - $e_{0,\mathrm{can}}$: For the canonical normalizations we get $e_{0,\mathrm{can}} = e_0 + (-0.014977... - 0.004866... \cdot i)e_1$.

index \ genus	0	1
7	3/3	0
8	1/1	0
9	9/9	1/1
10	9/9	1/1
11	6/6	0
12	23/27	2/3
13	22/23	1/1
14	21/29	1/2
15	54/62	7/9
16	36/65	7/9
17	16/35	1/2
total	200/269	21/28

TABLE 1. Number of computed passports that are currently in the database. (Note that there are no noncongruence subgroups with $\mu \leq 17$ and $g > 1$.)

$*$ $e_{1,\text{can}} = (0.059909... + 0.019465... \cdot i)e_1$.

(The weight six spaces are also given in the database, but are not listed here.)

4.5. **Status of the Database.** The number of computed passports in the current version can be found in Table 1. At the current stage the size of the database is $\sim$ 6GB in compressed and $\sim$ 16GB in uncompressed form. In total, about 25 000 hours of CPU time on `Intel Xeon E5-2680 v4 @ 2.40GHz` was used to compute the data.

4.6. **Reliability of the Results.** For genus zero, all coefficients have been computed using rigorous arithmetic over number fields or rigorous interval arithmetic. The only exception are the Eisenstein series whose precision we can only estimate heuristically. For higher genera, the coefficients are non-rigorous (but supported by highly convincing numerical evidence) because they have been guessed using the LLL algorithm. As an additional verification, we have also compared the numerical values of the rigorous expressions and the Eisenstein series to a computation using Hejhal's method with a different horocycle height to verify that the results match to at least 10 digits of precision.

4.7. **How to access the Database.** The database is currently available as a GITHUB repository at [2] and planned to be released to the LMFDB soon. The database entries can be loaded by running SAGE scripts which return a PYTHON dictionary containing the results. This results in more portability between different versions than storing the results as pickled objects. Additionally we provide the results in printed form as strings inside JSON files. To save on storage, we did not store the numerical approximations to the Fourier coefficients. For the same reason we also did not store the expressions over the number field L explicitly but instead generate them when loading the SAGE scripts by plugging in the values of v and w.

5. SOME NOTABLE EXAMPLES

5.1. **The largest degree of K.** The largest degree of K occurs for the passport 16_0_3_2_1_11_a for which we get $K = \mathbb{Q}(v)$ where $v = 1.531805... + 0.185685...i$ is a root of

$$0 = v^{24} - 6v^{23} + 21v^{22} - 60v^{21} + 184v^{20} - 478v^{19} + 651v^{18} - 1220v^{17} + 2230v^{16} + 1226v^{15} - 947v^{14} + 804v^{13} - 7092v^{12} - 6862v^{11} + 3971v^{10} - 15340v^{9} + 7975v^{8} + 36044v^{7} + 7134v^{6}$$

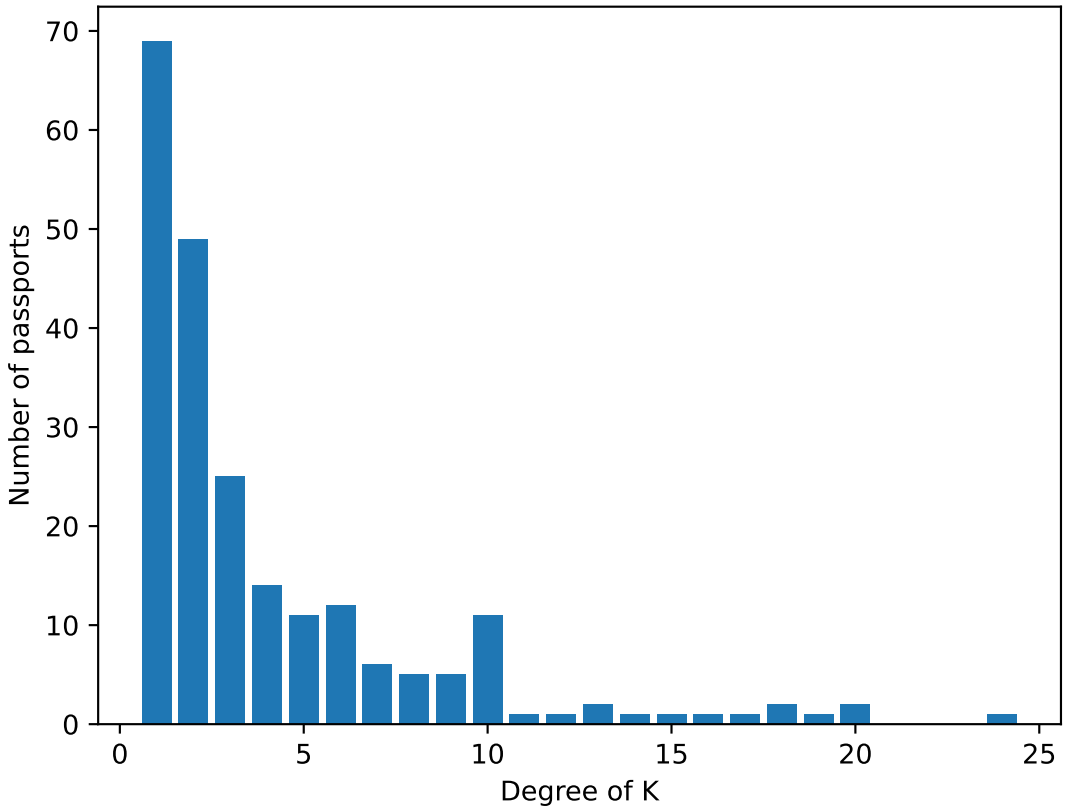

FIGURE 1. Distribution of the degrees of K in the database.

$$+ 14896v^5 + 13928v^4 - 2372v^3 + 3970v^2 - 584v + 22\,,$$

with discriminant $2^{52}3^{15}5^{10}7^{4}13^{4}$ and Galois group S_{24}. While it is definitely possible to use the method of [3, Section 5.1] to compute Belyi maps for higher number field degrees (see for example [19]), computing non-trivial amounts of Fourier coefficients becomes infeasible, which is the reason why these large passports have not been included in the database. To give an idea of the size of the coefficients involved, we remark that the first non-trivial coefficient of the Hauptmodul starts with

$$1803514480033338560116570995593169475394856903323733848361806547555555703 7...$$
$$...39589674217818100366320241631582723746983523118064516837 04v^{23} + ...\,,$$

which has 132 decimal digits and factors into

$$2^3 \cdot 3^4 \cdot 7^2 \cdot 13^3 \cdot 179 \cdot 1178062360621513$$
$$\cdot 151568799572553565849217592 1 \cdot 157310389634738553599688992620 03$$
$$\cdot 514197500928774955284304452843050396002488377491\,.$$

The distribution of degrees of K is given in Figure 1.

5.2. **Most Fourier expansion terms.** The largest number of Fourier expansion coefficients that have been computed for the database occurs for the passport 9_1_1_1_0_0_a for which we list 1459 coefficients. A distribution of the number of Fourier expansion terms is given in Figure 2. For genus zero subgroups we chose the number of terms based on some heuristic guesses, which depend on the degree of L, ensuring that the computation does not

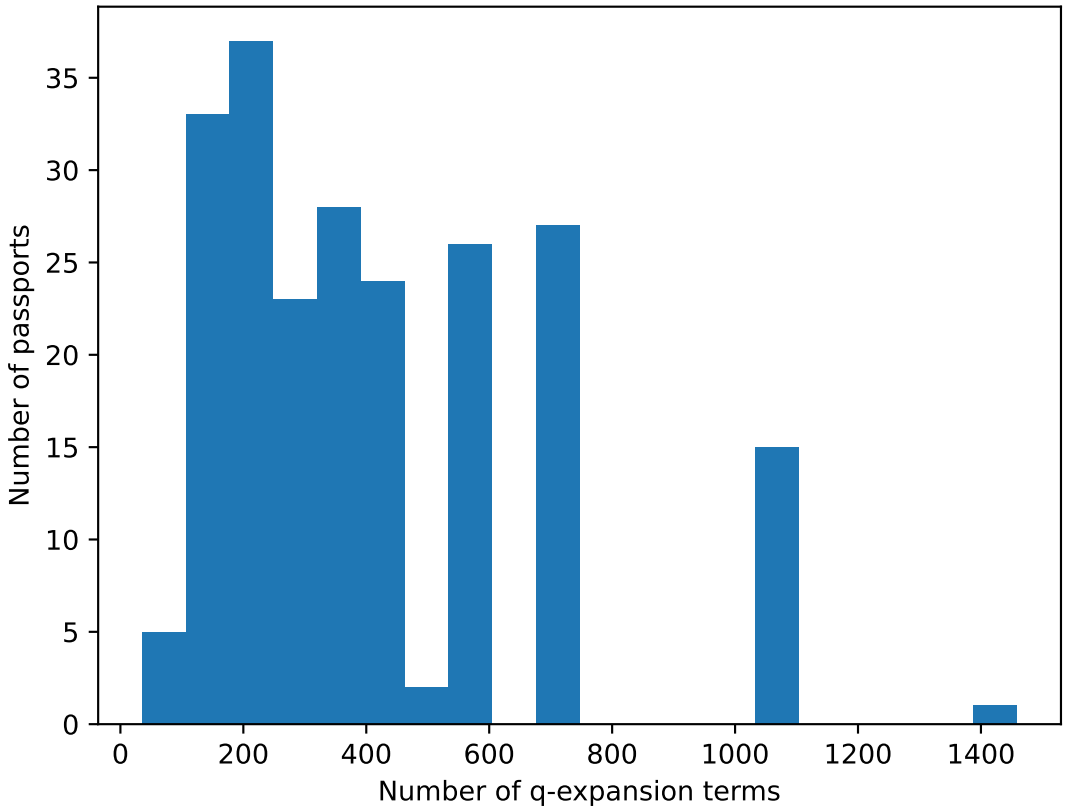

FIGURE 2. Distribution of the number of terms of the Fourier expansions in the database.

Passport Label	Elliptic Curve	Conductor
9_1_1_1_0_0_a	$y^2 + xy + y = x^3 - x^2 - 95x - 697$	162
10_1_1_0_1_0_a	$y^2 + xy + y = x^3 + x^2 - 110x - 880$	15
15_1_2_1_0_2_a	$y^2 + xy + y = x^3 - x^2 + 7x - 103$	270
15_1_2_1_0_3_b	$y^2 + xy = x^3 - x^2 - 41370x + 2022196$	2970

TABLE 2. Noncongruence subgroups corresponding to elliptic curves over $\mathbb{Q}$.

last longer than a few days. For higher genera subgroups we included as many terms as the LLL could determine reliably from the numerical approximations. For certain passports we also had to reduce the number of coefficients in order to satisfy the maximal GITHUB file size limit of 100MB.

5.3. **Elliptic curves defined over $\mathbb{Q}$.** In Table 2 we give all of the examples of genus 1 noncongruence subgroups from the database that correspond to elliptic curves over $\mathbb{Q}$, together with their defining equations and their conductors.

REFERENCES

[1] A. O. L. Atkin and H. P. F. Swinnerton-Dyer. Modular forms on noncongruence subgroups. In *Combinatorics (Proc. Sympos. Pure Math., Vol. XIX, Univ. California, Los Angeles, Calif., 1968)*, pages 1–25, 1971.

[2] D. Berghaus, H. Monien, and D. Radchenko. Database of modular forms of noncongruence subgroups of psl(2,zz). `https://github.com/David-Berghaus/noncong_database`, 2022. [Online; accessed 11 October 2022].

[3] D. Berghaus, H. Monien, and D. Radchenko. On the computation of modular forms on noncongruence subgroups, 2022.
[4] D. Berghaus and D. Radchenko. On the computation of eisenstein series on noncongruence subgroups, To be released.
[5] A. J. Best, J. Bober, A. R. Booker, E. Costa, J. E. Cremona, M. Derickx, M. Lee, D. Lowry-Duda, D. Roe, A. V. Sutherland, and J. Voight. Computing classical modular forms. In J. S. Balakrishnan, N. Elkies, B. Hassett, B. Poonen, A. V. Sutherland, and J. Voight, editors, *Arithmetic Geometry, Number Theory, and Computation*, pages 131–213, Cham, 2021. Springer International Publishing.
[6] F. Calegari, V. Dimitrov, and Y. Tang. The unbounded denominators conjecture, 2021.
[7] W. Y. Chen. Moduli interpretations for noncongruence modular curves. *Math. Ann.*, 371(1-2):41–126, 2018.
[8] H. Cohen and F. Strömberg. *Modular Forms: A Classical Approach*. Graduate Studies in Mathematics 179. American Mathematical Society, 2017.
[9] J. E. Cremona. *Algorithms for modular elliptic curves*. Cambridge University Press, Cambridge, second edition, 1997.
[10] F. Diamond and J. Shurman. *A First Course in Modular Forms*. Graduate Texts in Mathematics. Springer New York, 2006.
[11] S. Donnelly and J. Voight. A database of hilbert modular forms. In J. S. Balakrishnan, N. Elkies, B. Hassett, B. Poonen, A. V. Sutherland, and J. Voight, editors, *Arithmetic Geometry, Number Theory, and Computation*, pages 365–373, Cham, 2021. Springer International Publishing.
[12] L. Fang, J. W. Hoffman, B. Linowitz, A. Rupinski, and H. Verrill. Modular forms on noncongruence subgroups and Atkin-Swinnerton-Dyer relations. *Experiment. Math.*, 19(1):1–27, 2010.
[13] A. Fiori and C. Franc. The unbounded denominators conjecture for the noncongruence subgroups of index 7. *Journal of Number Theory*, 2022.
[14] F. Johansson. Arb: efficient arbitrary-precision midpoint-radius interval arithmetic. *IEEE Transactions on Computers*, 66:1281–1292, 2017.
[15] C. A. Kurth and L. Long. On modular forms for some noncongruence subgroups of $\mathrm{SL}_2(\mathbb{Z})$. *J. Number Theory*, 128(7):1989–2009, 2008.
[16] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261(4):515–534, 1982.
[17] W. Li, T. Liu, L. Long, R. Ramakrishna, and e. al. Noncongruence modular forms and modularity. `https://aimath.org/pastworkshops/noncongruence.html`, 2009. [Online; accessed 22 March 2022].
[18] M. H. Millington. Subgroups of the Classical Modular Group. *Journal of the London Mathematical Society*, s2-1(1):351–357, 01 1969.
[19] H. Monien. The sporadic group co3, hauptmodul and belyi map, 2018.
[20] M. Musty, S. Schiavone, J. Sijsling, and J. Voight. A database of Belyi maps. In *Proceedings of the Thirteenth Algorithmic Number Theory Symposium*, volume 2 of *Open Book Ser.*, pages 375–392. Math. Sci. Publ., Berkeley, CA, 2019.
[21] A. J. Scholl. Modular forms on noncongruence subgroups. In *Séminaire de Théorie des Nombres, Paris 1985–86*, volume 71 of *Progr. Math.*, pages 199–206. Birkhäuser Boston, Boston, MA, 1987.
[22] W. A. Stein et. al. Sage mathematics software (version 9.2). `https://www.sagemath.org/`, 2021. [Online; accessed 22 March 2022].
[23] F. Strömberg. Noncongruence subgroups and maass waveforms. `https://github.com/fredstro/noncongruence`, 2018. [Online; accessed 22 March 2022].
[24] F. Strömberg. Noncongruence subgroups and maass waveforms. *Journal of Number Theory*, 199:436–493, 2019.
[25] S. A. Vidal. An optimal algorithm to generate rooted trivalent diagrams and rooted triangular maps. *Theoret. Comput. Sci.*, 411(31-33):2945–2967, 2010.
[26] D. Zagier. Elliptic modular forms and their applications. In K. Ranestad, editor, *The 1-2-3 of Modular Forms: Lectures at a Summer School in Nordfjordeid, Norway*, pages 1–103. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

Bethe Center, University of Bonn, Nussallee 12, 53115 Bonn, Germany
Email address: `berghaus@th.physik.uni-bonn.de`

Bethe Center, University of Bonn, Nussallee 12, 53115 Bonn, Germany
Email address: `hmonien@uni-bonn.de`

Laboratoire Paul Painlevé, University of Lille, F-59655 Villeneuve d'Ascq, France
Email address: `danradchenko@gmail.com`