

Fast Computation of Generalized Dedekind Sums

Preston Tranbarger
 prestontranbarger@tamu.edu
 Department of Mathematics
 Texas A&M University
 College Station, TX 77843-3368, U.S.A.

Jessica Wang
 jwang22@wpi.edu
 Department of Mathematical Sciences
 Worcester Polytechnic Institute
 Worcester, MA 01609-2280, U.S.A.

Abstract

We construct an algorithm that reduces the complexity for computing generalized Dedekind sums from exponential to polynomial time. We do so by using an efficient word rewriting process in group theory.

1 Introduction and Main Result

The classical Dedekind sum is well-studied both inside and outside of number theory due to its connections with the Dedekind eta function and its applications in topology and combinatorial geometry. For more background on classical Dedekind sums, we refer the reader to [RG72].

Let h and k be coprime integers with $k > 0$. The classical Dedekind sum is defined as

$$s(h, k) = \sum_{n=1}^k B_1\left(\frac{n}{k}\right) B_1\left(\frac{hn}{k}\right),$$

where $B_1(x)$ is the first Bernoulli function

$$B_1(x) = \begin{cases} 0, & \text{if } x \in \mathbb{Z} \\ x - \lfloor x \rfloor - \frac{1}{2}, & \text{otherwise.} \end{cases}$$

The classical Dedekind sum satisfies the following reciprocity property ([RG72]):

$$s(h, k) = -s(k, h) + \frac{1}{12} \left(\frac{h}{k} + \frac{1}{hk} + \frac{k}{h} \right) - \frac{1}{4}.$$

Using the definition of classical Dedekind sum, it is readily seen that it can be computed in $O(k)$ time. However, one can obtain an $O(\log(k))$ time algorithm to compute the classical Dedekind sum using the reciprocity property ([BR15]).

The generalized Dedekind sum associated with newform Eisenstein series was introduced by Stucker, Vennos, and Young in [SVY20]. Since then, various aspects of generalized Dedekind sums have been studied, including the kernel ([NRY21], [LBVY21]), the image ([Maj22]), and their general behaviour ([DG20]).

Definition 1.1. Let $\gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma_0(q_1 q_2)$ with primitive Dirichlet characters χ_1, χ_2 and respective conductors q_1, q_2 . Let $q_1, q_2 > 1$ and $\chi_1 \chi_2(-1) = 1$, then

$$S_{\chi_1, \chi_2} \left(\begin{pmatrix} a & b \\ c & d \end{pmatrix} \right) = \sum_{j=1}^c \sum_{i=1}^{q_1} \left(\overline{\chi_2(j) \chi_1(i)} B_1 \left(\frac{j}{c} \right) B_1 \left(\frac{n}{q_1} + \frac{aj}{c} \right) \right).$$

The generalized Dedekind sum has the following crossed homomorphism property.

Lemma 1.2 (Crossed Homomorphism Property [SVY20]). *Let $\gamma_1, \gamma_2 \in \Gamma_0(q_1 q_2)$. Then*

$$S_{\chi_1, \chi_2}(\gamma_1 \gamma_2) = S_{\chi_1, \chi_2}(\gamma_1) + \psi(\gamma_1) S_{\chi_1, \chi_2}(\gamma_2),$$

where $\psi \left(\begin{pmatrix} a & b \\ c & d \end{pmatrix} \right) = \chi_1 \overline{\chi_2}(d)$.

Remark. In Lemma 1.2, $\psi(\gamma)$ is trivial on $\Gamma_1(q_1 q_2)$, so S_{χ_1, χ_2} may be viewed as an element of $\text{Hom}(\Gamma_1(q_1 q_2), \mathbb{C})$.

Note that it requires $O(cq_1)$ time to compute a generalized Dedekind sum from the definition. Similar to the classical Dedekind sums, we are interested in constructing a faster algorithm. Instead of using the reciprocity property, we provide an alternative approach using a word-rewriting process.

Theorem 1.3. *Given primitive Dirichlet characters χ_1, χ_2 and respective conductors $q_1, q_2 > 1$ such that $\chi_1 \chi_2(-1) = 1$. Let $\gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma_0(q_1 q_2)$. For fixed q_1, q_2 , the time complexity of finding $S_{\chi_1, \chi_2}(\gamma)$ as a function of γ is $O(\log(c))$.*

Remark. The algorithm for Theorem 1.3 can be found in Section 3.1. We give the specific details of our model of computation in Section 3.2.

Before diving into the technicalities of the algorithm, we provide the reader with a general outline. Given $\gamma \in \Gamma_1(q_1 q_2) < \text{SL}_2(\mathbb{Z})$ written as a word in the generators of $\text{SL}_2(\mathbb{Z})$, we can apply the Reidemeister rewriting process to express it as a word in the elements of a particular generating set of $\Gamma_1(q_1 q_2)$. By precomputing the Dedekind sum of each element of this generating set, we can use Lemma 1.2 to compute any Dedekind sum. However, in using the Reidemeister rewriting process, the length of the word can be exponentially large in terms of the logarithms of the entries of γ . Therefore, to achieve the polynomial time in Theorem 1.3, we modify the Reidemeister rewriting process, as in Theorem 2.10 below, to collect the exponents of successive letters in the rewritten word. In the specific case of $\Gamma_1(q_1 q_2)$, we develop a useful identity in Lemma 2.15 to ensure a finite alphabet.

2 Preliminaries

2.1 General Preliminaries

In this section we will define some general group theoretic definitions and results which will aid in the construction of the algorithm. For the rest of this subsection, we let G be a finitely generated group and H be a subgroup of G . We will work with specific groups in Section 2.2.

Definition 2.1. *We say \mathcal{T} is a right transversal of H in G if each right coset of H in G contains exactly one element of \mathcal{T} . Moreover, \mathcal{T} must contain the identity.*

Note that a transversal differs from an arbitrary set of coset representatives in that it must contain an identity. This fact proves to be essential for later preliminaries.

Definition 2.2. *Given a right transversal \mathcal{T} of H in G , a right coset representative function for \mathcal{T} is a mapping: $G \rightarrow \mathcal{T}$ via $g \mapsto \bar{g}$, where \bar{g} is the unique element in \mathcal{T} such that $Hg = H\bar{g}$.*

We present a simple lemma which will be used repeatedly throughout this paper.

Lemma 2.3. *Given a right transversal of H in G and $a, b \in G$,*

$$\overline{ab} = \overline{a}\overline{b}.$$

Proof. By Definition 2.2, $H(\overline{ab}) = H(ab) = (Ha)b = (H\bar{a})b = H(\overline{a}\overline{b}) = H(\overline{ab})$. \square

We continue by defining an important function and exploring some of its properties.

Definition 2.4. *Given a right transversal of H in G and $a, b \in G$, we define*

$$U(a, b) = ab(\overline{ab})^{-1}.$$

Lemma 2.5. *Given a right transversal of H in G and $a, b \in G$, then $U(a, b) \in H$.*

Proof. By Definition 2.2, $Hab = H\overline{ab}$, thus $Hab(\overline{ab})^{-1} = H$, so $ab(\overline{ab})^{-1} \in H$. \square

Given a finite set of generators for a group, we use the information thus far to describe a set of generators for a given subgroup.

Lemma 2.6 (Schreier's Lemma [MKS04, Theorem 2.7]). *Let \mathcal{S} be a set which finitely generates G , and let \mathcal{T} be a right transversal of H in G . The set of Schreier generators*

$$\{U(t, s) : t \in \mathcal{T}, s \in \mathcal{S}\}$$

generates H .

Remark. *We say a set generates a group if every element in the group can be expressed as a combination of elements in the set and their inverses.*

We now describe a rewriting process.

Theorem 2.7 (Reidemeister Rewriting Process [MKS04, Corollary 2.7.2]). *Let $G = \langle g_1, \dots, g_n \rangle$. Let $h = g_{q_1}^{\epsilon_1} g_{q_2}^{\epsilon_2} \cdots g_{q_r}^{\epsilon_r} \in H$ (where $\epsilon_k = \pm 1$) be a word in the g_i . Fix a right transversal of H in G . Define the mapping τ of the word h by*

$$\tau(h) = U(p_1, g_{q_1})^{\epsilon_1} U(p_2, g_{q_2})^{\epsilon_2} \cdots U(p_r, g_{q_r})^{\epsilon_r},$$

where

$$p_k = \begin{cases} \overline{g_{q_1}^{\epsilon_1} g_{q_2}^{\epsilon_2} \cdots g_{q_{k-1}}^{\epsilon_{k-1}}} & \text{if } \epsilon_k = 1 \\ g_{q_1}^{\epsilon_1} g_{q_2}^{\epsilon_2} \cdots g_{q_k}^{\epsilon_k} & \text{if } \epsilon_k = -1. \end{cases}$$

Then $\tau(h) = h$, for all $h \in H$.

The Reidemeister rewriting process allows us to express a word in the generators of G as a word in the Schreier generators of H (using Lemma 2.6).

Example 2.8. Let $G = \langle g_1, \dots, g_n \rangle$, and H be a subgroup of G . Let $h = g_1 g_1 g_1 g_2^{-1} g_2^{-1} \in H$, then by Theorem 2.7,

$$\tau(h) = U(\overline{1}, g_1)U(\overline{g_1}, g_1)U(\overline{g_1^2}, g_1)U(\overline{g_1^3g_2^{-1}}, g_2)^{-1}U(\overline{g_1^3g_2^{-2}}, g_2)^{-1} = h. \quad (1)$$

Note that Theorem 2.7 requires $\epsilon_k = \pm 1$ (for example, h must be written as $g_1 g_1 g_1 g_2^{-1} g_2^{-1}$, not $g_1^3 g_2^{-2}$). Since the length of $\tau(h)$ is the same as the length of h , this rewriting process is often inefficient. We provide Theorem 2.10 to reduce the number of computations, which requires the lemma below.

Lemma 2.9. Let $a, b \in G$ and $k \in \mathbb{Z}_{>0}$. Given a right transversal of H in G , the following product identities hold:

$$U(\bar{a}, b^k) = U(\bar{a}, b)U(\bar{ab}, b)\dots U(\bar{ab}^{k-1}, b), \quad (2)$$

$$U(\bar{a}, b^{-k}) = U(\bar{ab^{-1}}, b)^{-1} U(\bar{ab^{-2}}, b)^{-1} \dots U(\bar{ab^{-k}}, b)^{-1}. \quad (3)$$

Proof. By Definition 2.4 the right hand side of (2) equals

$$\overline{ab} \left(\overline{ab} \right)^{-1} \overline{abb} \left(\overline{abb} \right)^{-1} \dots \overline{ab^{k-1}b} \left(\overline{ab^{k-1}b} \right)^{-1}. \quad (4)$$

Applying Lemma 2.3, this simplifies to

$$\bar{a}b(\bar{a}b)^{-1}\bar{a}b(\bar{a}b^2)^{-1}\dots\bar{a}b^{k-1}b\left(\bar{a}b^k\right)^{-1}. \quad (5)$$

Note that many terms cancel, so (5) becomes $\bar{a}b^k(\bar{a}b^k)^{-1} = U(\bar{a}, b^k)$. The proof of (3) follows in a similar manner, though care is required in handling the inverses. \square

Theorem 2.10 (Modified Reidemeister Rewriting Process). *Given a right transversal of H in G , let $G = \langle g_1, \dots, g_n \rangle$. Let $h = g_{q_1}^{a_1} g_{q_2}^{a_2} \cdots g_{q_r}^{a_r} \in H$ (where $a_i \in \mathbb{Z}_{\neq 0}$) be a word in powers of the g_i . Define the mapping τ of the word h by*

$$\tau(h) = U(p_1, g_{q_1}^{a_1})U(p_2, g_{q_2}^{a_2}) \cdots U(p_r, g_{q_r}^{a_r}),$$

where

$$p_k = \overline{g_{q_1}^{a_1} g_{q_2}^{a_2} \cdots g_{q_{k-1}}^{a_{k-1}}}.$$

Then $\tau(h) = h$, for all $h \in H$.

Proof. This follows by applying Lemma 2.9 to Theorem 2.7.

We illustrate Theorem 2.10 and its proof by an example.

Example 2.11. Continuing with the assumptions in Example 2.8, we now write $h = g_1 g_1 g_2 g_2^{-1} g_2^{-1} = g_1^3 g_2^{-2} \in H$. We want to show (as Theorem 2.10 claims) that $\tau(h) = U(\bar{1}, g_1^3)U(\bar{g_1^3}, g_2^{-2})$. From Lemma 2.9, we have

$$U(\overline{1}, g_1^3) = U(\overline{1}, g_1)U(\overline{g_1}, g_1)U(\overline{g_1^2}, g_1),$$

$$U(\overline{g_1^3}, g_2^{-2}) = U(\overline{g_1^3g_2^{-1}}, g_2)^{-1}U(\overline{g_1^3g_2^{-2}}, g_2)^{-1}.$$

So (1) becomes

$$\tau(h) = U(\bar{1}, g_1^3)U(\bar{g_1^3}, g_2^{-2}) = h,$$

which is in the form of Theorem 2.10.

As desired, this process provides us with a product expansion with far fewer terms than that of Theorem 2.7.

2.2 Specific Preliminaries

Let us now consider the subgroup $\Gamma_1(N)$ of $\mathrm{SL}_2(\mathbb{Z})$.

Definition 2.12. *Let*

$$S = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

The following lemma is well known.

Lemma 2.13 ([Iwa97, Theorem 1.1]). *We have*

$$\mathrm{SL}_2(\mathbb{Z}) = \langle S, T \rangle.$$

More specifically, any matrix $M \in \mathrm{SL}_2(\mathbb{Z})$ can be decomposed into the following form:

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \pm T^{a_1} S T^{a_2} S \dots T^{a_{r-1}} S T^{a_r}. \quad (6)$$

Note that $-I = S^2$.

Remark. One can quickly compute the values of the a_i via a variant on the Euclidean algorithm.

Remark. In (6), the sum of the $|a_i|$ grows as $O(c)$ but r grows as $O(\log(c))$ (this follows as a consequence of the above remark and [Knu97, Section 4.5.3]). This is significant because the original Reidemeister rewriting process treats M as a word in S and T with length $(|a_1| + \dots + |a_r|) + r = O(c)$. The modified Reidemeister rewriting process treats M as a word in the powers of S and T with length $2r$ which grows as $O(\log(c))$.

In the case of $\Gamma_1(N)$ in $\mathrm{SL}_2(\mathbb{Z})$, the modified Reidemeister rewriting process allows us to express a word in the generators of $\mathrm{SL}_2(\mathbb{Z})$ as a word in the elements of $\Gamma_1(N)$ with the form $U(l, g^k)$, where l is in a right transversal of $\Gamma_1(N)$ in $\mathrm{SL}_2(\mathbb{Z})$ and $g \in \{S, T\}$ and $k \in \mathbb{Z}$. However, since k is arbitrarily large, this set of elements is infinite. We present a way to reduce this infinite set to a finite set.

Lemma 2.14. *Given a right transversal of $\Gamma_1(N)$ in $\mathrm{SL}_2(\mathbb{Z})$,*

$$\overline{MT^N} = \overline{M}$$

for all $M \in \mathrm{SL}_2(\mathbb{Z})$.

Proof. Note $T^N \in \Gamma(N)$. Since $\Gamma(N)$ is normal in $\mathrm{SL}_2(\mathbb{Z})$, we have $MT^N M^{-1} \in \Gamma(N) < \Gamma_1(N)$ for all $M \in \mathrm{SL}_2(\mathbb{Z})$. Thus $\Gamma_1(N)MT^N = \Gamma_1(N)M$ and $\overline{MT^N} = \overline{M}$. \square

Lemma 2.15. *Let $a = qN + r$ for $0 \leq r < N$ and let $M \in \mathrm{SL}_2(\mathbb{Z})$. Given a right transversal of $\Gamma_1(N)$ in $\mathrm{SL}_2(\mathbb{Z})$,*

$$U(\overline{M}, T^a) = U^q(\overline{M}, T^N)U(\overline{M}, T^r).$$

Proof. Beginning with Definition 2.4, we have

$$U(\overline{M}, T^a) = \overline{M} T^a \left(\overline{M} T^a \right)^{-1}. \quad (7)$$

By applying Lemma 2.14, expanding T^a , and multiplying by the identity $I = (\overline{M})^{-1} \overline{M}$, (7) becomes

$$\overline{M} T^{qN} (\overline{M})^{-1} \overline{M} T^r (\overline{M} T^r)^{-1}.$$

By Lemmas 2.3 and 2.14, $\overline{M} = \overline{\overline{M}} = \overline{M} T^N$. Since $\overline{M} T^{qN} (\overline{M})^{-1} = (\overline{M} T^N (\overline{M})^{-1})^q$, we get that (7) equals

$$\left(\overline{M} T^N \left(\overline{M} T^N \right)^{-1} \right)^q \overline{M} T^r \left(\overline{M} T^r \right)^{-1}. \quad \square$$

Using Lemma 2.15 in conjunction with Theorem 2.10, we can rewrite every word in the generators of $\mathrm{SL}_2(\mathbb{Z})$ as a word in the letters of a finite subset of $\Gamma_1(N)$. This is central to our algorithm in Section 3.1. However, before we are able to present this algorithm, we need a few more results describing the structure of congruence subgroups of $\mathrm{SL}_2(\mathbb{Z})$. One can find these results in many sources, including [Ste07].

Lemma 2.16. *We have*

$$[\Gamma_1(N) : \Gamma_0(N)] = N \prod_{p|N} \left(1 - \frac{1}{p} \right),$$

$$[\Gamma_0(N) : \mathrm{SL}_2(\mathbb{Z})] = N \prod_{p|N} \left(1 + \frac{1}{p} \right),$$

$$[\Gamma_1(N) : \mathrm{SL}_2(\mathbb{Z})] = N^2 \prod_{p|N} \left(1 - \frac{1}{p^2} \right).$$

Lemma 2.17. *There exists a bijection $\Gamma_1(N) \setminus \Gamma_0(N) \rightarrow (\mathbb{Z}/N\mathbb{Z})^\times$ via*

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \mapsto d \bmod N.$$

Lemma 2.18 ([Ste07, Proposition 8.6]). *Let $P = \{(c, d) : c, d \in \mathbb{Z}/N\mathbb{Z}, \gcd(c, d, N) = 1\}$. There exists a bijection $\Gamma_1(N) \setminus \mathrm{SL}_2(\mathbb{Z}) \rightarrow P$ via*

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \mapsto (c \bmod N, d \bmod N).$$

3 Algorithm

In this section, we provide an algorithm for computing the generalized Dedekind sums and a time complexity analysis for each step of the algorithm. We also compare our algorithm to the naive algorithm of simply using Definition 1.1. We divide our algorithm into precomputations (which only needs to be computed once for each pair of characters) and the main computation.

3.1 Stating the Algorithm

Let $N = q_1 q_2$ for primitive Dirichlet characters χ_1, χ_2 with respective conductors q_1, q_2 . Let $q_1, q_2 > 1$ and $\chi_1 \chi_2(-1) = 1$. Given $\gamma_0 = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma_0(N)$, we present an algorithm to find $S_{\chi_1, \chi_2}(\gamma_0)$.

Group Theoretic Precomputation

- Find a right transversal \mathcal{T}_{Γ_0} of $\Gamma_1(N)$ in $\Gamma_0(N)$ (using Lemma 2.17).
- Find a right transversal $\mathcal{T}_{\text{SL}_2(\mathbb{Z})}$ of $\Gamma_1(N)$ in $\text{SL}_2(\mathbb{Z})$ (using Lemma 2.18).
- Find the set $\mathcal{U} = \{U(t, T^i) : t \in \mathcal{T}_{\text{SL}_2(\mathbb{Z})}, 1 \leq i \leq N\} \cup \{U(t, S^k) : t \in \mathcal{T}_{\text{SL}_2(\mathbb{Z})}, 0 \leq k \leq 2\}$. Note this set includes the set of the Schreier generators of $\Gamma_1(N)$ in $\text{SL}_2(\mathbb{Z})$ (see Lemma 2.6).

Dedekind Sum Precomputation

- Use Definition 1.1 to compute the Dedekind sums $S_{\chi_1, \chi_2}(\mathcal{T}_{\Gamma_0})$.
- Use Definition 1.1 to compute the Dedekind sums $S_{\chi_1, \chi_2}(\mathcal{U})$.

The Main Computation

We write $\gamma_0 = \gamma_1 g$, where $\gamma_1 \in \Gamma_1(N)$ and $g \in \mathcal{T}_{\Gamma_0}$. Let $g \mapsto \bar{g}$ denote the right coset representative function uniquely described by $\mathcal{T}_{\text{SL}_2(\mathbb{Z})}$ per Definition 2.2. By Lemma 1.2,

$$S_{\chi_1, \chi_2}(\gamma_0) = S_{\chi_1, \chi_2}(\gamma_1) + S_{\chi_1, \chi_2}(g).$$

Since $g \in \mathcal{T}_{\Gamma_0}$, $S_{\chi_1, \chi_2}(g)$ has been precomputed, so we are now only concerned with $S_{\chi_1, \chi_2}(\gamma_1)$. Using Lemma 2.13, we write

$$\gamma_1 = \pm T^{a_1} S T^{a_2} S \dots T^{a_{r-1}} S T^{a_r}.$$

Using Theorem 2.10, we rewrite

$$\tau(\gamma_1) = U(\overline{p_1}, T^{a_1}) U(\overline{p_1 T^{a_1}}, S) U(\overline{p_2}, T^{a_2}) U(\overline{p_2 T^{a_2}}, S) \dots U(\overline{p_r}, T^{a_r}) U(\overline{p_r T^{a_r}}, \pm I) = \gamma_1, \quad (8)$$

where

$$p_k = T^{a_1} S T^{a_2} S \dots T^{a_{k-1}} S.$$

Now we apply Lemma 2.15. For each exponent of T , we write $a_i = q_i N + r_i$ with $0 \leq r_i < N$. Then

$$U(\overline{p_i}, T^{a_i}) = U^{q_i}(\overline{p_i}, T^N) U(\overline{p_i}, T^{r_i}). \quad (9)$$

We apply the Dedekind sum to (9). Since $\mathcal{U} \subset \Gamma_1(N)$, by Lemma 1.2,

$$S_{\chi_1, \chi_2}(U(\overline{p_i}, T^a)) = q_i S_{\chi_1, \chi_2}(U(\overline{p_i}, T^N)) + S_{\chi_1, \chi_2}(U(\overline{p_i}, T^{r_i})). \quad (10)$$

Using (8) and (9), we can express γ_1 as a product of elements in \mathcal{U} . Applying Lemma 1.2 to $S_{\chi_1, \chi_2}(\tau(\gamma_1))$ and expanding via (10), we acquire the desired Dedekind sum, each term of which has been precomputed.

Remark. Given $\mathcal{T}_{\text{SL}_2(\mathbb{Z})}$ and $g \in \text{SL}_2(\mathbb{Z})$, we can use Proposition 2.18 to determine \bar{g} .

Example Computation

Consider $\Gamma_0(9)$. Let $\chi_1 = \chi_2$ be the primitive character modulo 3 with conductors $q_1 = q_2 = 3$. We want to compute $S_{\chi_1, \chi_2}(\gamma_0)$ where

$$\gamma_0 = \begin{pmatrix} 17 & 32 \\ 9 & 17 \end{pmatrix}.$$

From the precomputations, we acquire a right transversal of $\Gamma_1(9)$ in $\Gamma_0(9)$

$$\mathcal{T}_{\Gamma_0} = \left\{ \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), \left(\begin{smallmatrix} 5 & 1 \\ 9 & 2 \end{smallmatrix} \right), \left(\begin{smallmatrix} 7 & 3 \\ 9 & 4 \end{smallmatrix} \right), \left(\begin{smallmatrix} 2 & 1 \\ 9 & 5 \end{smallmatrix} \right), \left(\begin{smallmatrix} 4 & 3 \\ 9 & 7 \end{smallmatrix} \right), \left(\begin{smallmatrix} 8 & 7 \\ 9 & 8 \end{smallmatrix} \right) \right\},$$

a right transversal of $\Gamma_1(9)$ in $\mathrm{SL}_2(\mathbb{Z})$

$$\mathcal{T}_{\mathrm{SL}_2(\mathbb{Z})} = \left\{ \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), \left(\begin{smallmatrix} 5 & 1 \\ 9 & 2 \end{smallmatrix} \right), \left(\begin{smallmatrix} 7 & 3 \\ 9 & 4 \end{smallmatrix} \right), \left(\begin{smallmatrix} 2 & 1 \\ 9 & 5 \end{smallmatrix} \right), \left(\begin{smallmatrix} 4 & 3 \\ 9 & 7 \end{smallmatrix} \right), \dots, \left(\begin{smallmatrix} 5 & 8 \\ 8 & 13 \end{smallmatrix} \right), \left(\begin{smallmatrix} 5 & 3 \\ 8 & 5 \end{smallmatrix} \right), \left(\begin{smallmatrix} 7 & 13 \\ 8 & 15 \end{smallmatrix} \right), \left(\begin{smallmatrix} 7 & 6 \\ 8 & 7 \end{smallmatrix} \right), \left(\begin{smallmatrix} 1 & 2 \\ 8 & 17 \end{smallmatrix} \right) \right\},$$

and the set

$$\mathcal{U} = \{U(t, T^i) : t \in \mathcal{T}_{\mathrm{SL}_2(\mathbb{Z})}, 1 \leq i \leq 9\} \cup \{U(t, S^k) : t \in \mathcal{T}_{\mathrm{SL}_2(\mathbb{Z})}, 0 \leq k \leq 2\}.$$

From the precomputations, we also acquire the Dedekind sums $S_{\chi_1, \chi_2}(\mathcal{T}_{\Gamma_0})$ and $S_{\chi_1, \chi_2}(\mathcal{U})$. By Lemma 2.17, we can write $\gamma_0 = \gamma_1 g$, where

$$\gamma_1 = \begin{pmatrix} -152 & 137 \\ -81 & 73 \end{pmatrix} \in \Gamma_1(9) \quad \text{and} \quad g = \begin{pmatrix} 8 & 7 \\ 9 & 8 \end{pmatrix} \in \mathcal{T}_{\Gamma_0}.$$

Since $g \in \mathcal{T}_{\Gamma_0}$, $S_{\chi_1, \chi_2}(g)$ has been precomputed, so now we only need to compute $S_{\chi_1, \chi_2}(\gamma_1)$. Using Lemma 2.13, we compute

$$\gamma_1 = -T^1 S T^{-2} S T^{-2} S T^{-2} S T^{-2} S T^{-2} S T^{-2} S T^{-11} S T^{-1}. \quad (11)$$

Applying Theorem 2.10 to (11) with all p_i written in matrix forms, we get

$$\begin{aligned} \tau(\gamma_1) = & U\left(\overline{\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right)}, T^1\right) U\left(\overline{\left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix} \right)}, S\right) U\left(\overline{\left(\begin{smallmatrix} 1 & -1 \\ 1 & 0 \end{smallmatrix} \right)}, T^{-2}\right) U\left(\overline{\left(\begin{smallmatrix} 1 & -3 \\ 1 & -2 \end{smallmatrix} \right)}, S\right) \dots \\ & \dots U\left(\overline{\left(\begin{smallmatrix} -15 & -13 \\ -8 & -7 \end{smallmatrix} \right)}, T^{-11}\right) U\left(\overline{\left(\begin{smallmatrix} -15 & 152 \\ -8 & 81 \end{smallmatrix} \right)}, S\right) U\left(\overline{\left(\begin{smallmatrix} 152 & 15 \\ 81 & 8 \end{smallmatrix} \right)}, T^{-1}\right) U\left(\overline{\left(\begin{smallmatrix} 152 & -137 \\ 81 & -73 \end{smallmatrix} \right)}, -I\right) = \gamma_1. \end{aligned}$$

Applying Lemmas 2.15 and 2.18 to each term of the above product, we get the following computation.

$$\begin{aligned} U\left(\overline{\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right)}, T^1\right) &= U^0\left(\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), T^9\right) U\left(\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), T^1\right), & U\left(\overline{\left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix} \right)}, S\right) &= U\left(\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), S\right), \\ U\left(\overline{\left(\begin{smallmatrix} 1 & -1 \\ 1 & 0 \end{smallmatrix} \right)}, T^{-2}\right) &= U^{-1}\left(\left(\begin{smallmatrix} 1 & 8 \\ 1 & 9 \end{smallmatrix} \right), T^9\right) U\left(\left(\begin{smallmatrix} 1 & 8 \\ 1 & 9 \end{smallmatrix} \right), T^7\right), & U\left(\overline{\left(\begin{smallmatrix} 1 & -3 \\ 1 & -2 \end{smallmatrix} \right)}, S\right) &= U\left(\left(\begin{smallmatrix} 1 & 6 \\ 1 & 7 \end{smallmatrix} \right), S\right), \\ & \vdots \\ U\left(\overline{\left(\begin{smallmatrix} -15 & -13 \\ -8 & -7 \end{smallmatrix} \right)}, T^{-11}\right) &= U^{-2}\left(\left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix} \right), T^9\right) U\left(\left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix} \right), T^7\right), & U\left(\overline{\left(\begin{smallmatrix} -15 & 152 \\ -8 & 81 \end{smallmatrix} \right)}, S\right) &= U\left(\left(\begin{smallmatrix} 1 & 8 \\ 1 & 9 \end{smallmatrix} \right), S\right), \\ U\left(\overline{\left(\begin{smallmatrix} 152 & 15 \\ 81 & 8 \end{smallmatrix} \right)}, T^{-1}\right) &= U^{-1}\left(\left(\begin{smallmatrix} 8 & 7 \\ 9 & 8 \end{smallmatrix} \right), T^9\right) U\left(\left(\begin{smallmatrix} 8 & 7 \\ 9 & 8 \end{smallmatrix} \right), T^8\right), & U\left(\overline{\left(\begin{smallmatrix} 152 & -137 \\ 81 & -73 \end{smallmatrix} \right)}, -I\right) &= U\left(\left(\begin{smallmatrix} 8 & 7 \\ 9 & 8 \end{smallmatrix} \right), S^2\right). \end{aligned}$$

Note that every term on the right hand side of these equalities are in the precomputed set \mathcal{U} . Thus, using Lemma 1.2,

$$\begin{aligned} S_{\chi_1, \chi_2}(\gamma_1) &= 0 \cdot S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), T^9\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), T^1\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right), S\right)) \\ &\quad - 1 \cdot S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 8 \\ 1 & 9 \end{smallmatrix} \right), T^9\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 8 \\ 1 & 9 \end{smallmatrix} \right), T^7\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 6 \\ 1 & 7 \end{smallmatrix} \right), S\right)) \\ &\quad \vdots \\ &\quad - 2 \cdot S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix} \right), T^9\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix} \right), T^7\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 1 & 8 \\ 1 & 9 \end{smallmatrix} \right), S\right)) \\ &\quad - 1 \cdot S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 8 & 7 \\ 9 & 8 \end{smallmatrix} \right), T^9\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 8 & 7 \\ 9 & 8 \end{smallmatrix} \right), T^8\right)) + S_{\chi_1, \chi_2}(U\left(\left(\begin{smallmatrix} 8 & 7 \\ 9 & 8 \end{smallmatrix} \right), S^2\right)). \end{aligned}$$

Now, using the precomputed Dedekind sums, $S_{\chi_1, \chi_2}(\gamma_0) = S_{\chi_1, \chi_2}(\gamma_1) + S_{\chi_1, \chi_2}(g) = 0$.

Remark. This example shows that $\gamma_0 = \begin{pmatrix} 17 & 32 \\ 9 & 17 \end{pmatrix}$ lies in the kernel of this Dedekind sum. For more information on the kernel of Dedekind sums, we refer the reader to [NRY21, LVB21].

3.2 Analysis of Algorithm

First we discuss a simplified model of our computation. We consider a matrix multiplication as one operation. Since we work in $\mathrm{SL}_2(\mathbb{Z})$, we assume computing the inverse of a matrix takes constant time. Since the Dedekind sums lie in cyclotomic extensions of \mathbb{Q} , we can represent any Dedekind sum as a linear combination of powers of a given root of unity. Adding and multiplying these linear combinations is rather trivial, so we assume that addition and multiplication of Dedekind sums take constant time.

Lemma 3.1. *Given $g \in \mathrm{SL}_2(\mathbb{Z})$ and the right transversals $\mathcal{T}_{\mathrm{SL}_2(\mathbb{Z})}$ or \mathcal{T}_{Γ_0} , finding \bar{g} under the right coset representative function for $\Gamma_1(N) \backslash \Gamma_0(N)$ or $\Gamma_1(N) \backslash \mathrm{SL}_2(\mathbb{Z})$ requires $O(1)$ time.*

Proof. We simply use the bijections stated in Lemmas 2.17 and 2.18 to find the transversals. \square

Proposition 3.2. *In Section 3.1, the time complexity of the group theoretic precomputations is $O(N^3)$.*

Proof. The most computationally expensive step is finding the set $\mathcal{U} = \{U(t, T^i) : t \in \mathcal{T}_{\mathrm{SL}_2(\mathbb{Z})}, 1 \leq i \leq N\} \cup \{U(t, S^k) : t \in \mathcal{T}_{\mathrm{SL}_2(\mathbb{Z})}, 0 \leq k \leq 2\}$. This set has at most $(N+3)|\mathcal{T}_{\mathrm{SL}_2(\mathbb{Z})}|$ elements, thus $|\mathcal{U}|$ grows as $O(N^3)$. Since we define matrix multiplication as one operation and determining the transversal of an element under the right coset representative function takes $O(1)$ time (by Lemma 3.1), it follows that the total time to compute the elements of \mathcal{U} takes $O(N^3)$ time. \square

Proposition 3.3. *Given $\gamma \in \Gamma_0(N)$, the time complexity of the Dedekind sum precomputations for finding $S_{\chi_1, \chi_2}(\mathcal{T}_{\Gamma_0} \cup \mathcal{U})$ is $O(N^3 C q_1)$, where C denotes the maximum absolute value of the lower-left entry of the elements in $\mathcal{T}_{\Gamma_0} \cup \mathcal{U}$.*

Note that C is solely dependent on the value of N .

Proof. Using Lemma 2.16, we have

$$[\Gamma_1(N) : \Gamma_0(N)] + |\mathcal{U}| = N \left(\prod_{p|N} \left(1 - \frac{1}{p}\right) \right) + N^2(N+3) \left(\prod_{p|N} \left(1 - \frac{1}{p^2}\right) \right) = O(N^3).$$

Since it takes $O(cq_1)$ steps to compute the Dedekind sum of a matrix with lower-left entry c from Definition 1.1, it takes $O(N^3 C q_1)$ steps to compute the Dedekind sum of all elements in $\mathcal{T}_{\Gamma_0} \cup \mathcal{U}$. \square

Definition 3.4. *Let $\gamma = \pm T^{a_1} S T^{a_2} S \cdots T^{a_k} \in \mathrm{SL}_2(\mathbb{Z})$. We say T^{a_i} or S is a letter, and $2k$ is the length of γ .*

Recall that Theorem 1.3 states that the main computation algorithm provided in Section 3.1 has time complexity $O(\log(c))$. We provide a proof below.

Proof of Theorem 1.3. We claim that the time complexity for the modified Reidemeister rewriting process (Theorem 2.10) is $O(\log(c))$. In (8), given p_i , each p_{i+1} takes one operation to compute (through multiplying by a matrix on the right). We know that $2k$ grows in $O(\log(c))$ due to Lemma 2.13, hence it takes $O(\log(c))$ steps to compute all p_i . By Lemma 3.1 given $g \in \mathrm{SL}_2(\mathbb{Z})$, finding

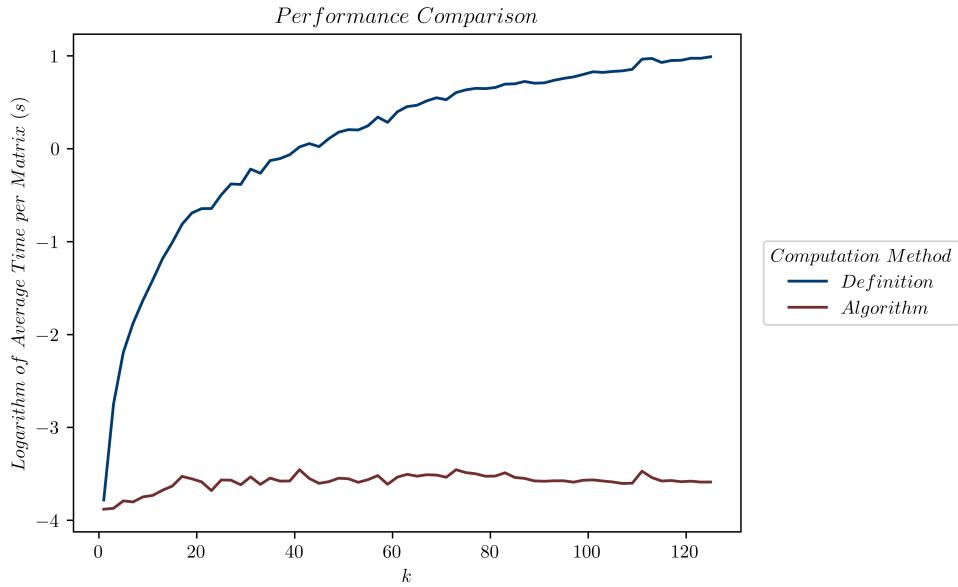
the corresponding element \bar{g} in \mathcal{T}_{Γ_0} has complexity $O(1)$. Hence in (8), each of the U -functions takes $O(1)$ time to compute. Since γ_1 has length $2k$, it also takes $O(\log(c))$ steps to compute the U -functions given p_i . Hence the time complexity for the modified Reidemeister rewriting process is $O(\log(c)) + O(\log(c)) = O(\log(c))$.

Note that reducing the power of a generator in (9) using Lemma 2.15 and looking up the precomputed Dedekind sum of each letter both have complexity $O(1)$, which does not affect the time complexity of the algorithm. \square

3.3 Comparison of Algorithms

In this section, we give some experimental evidence for the speed of our algorithm in comparison to that which uses Definition 1.1 (using the implementations found in Section 3.4)

Example 3.5. Consider $\Gamma_0(28)$. Let χ_1 be the primitive Dirichlet character with conductor $q_1 = 4$, and let χ_2 be the primitive Dirichlet character with conductor $q_2 = 7$ such that $\chi_2(3) = \exp(2\pi i(5/6))$. We let $\gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ where $c = 28k$, $0 < a < c$, and $\gcd(a, c) = 1$. We choose b and d such that the exponent a_r is 0 after applying Lemma 2.13. We compute the Dedekind sum $S_{\chi_1, \chi_2}(\gamma)$ of all matrices that satisfy the conditions, and graph the logarithm of the average time it takes to compute each k .



Note that the performance of the algorithm always exceeded that of the definition for this pair of characters.

Example 3.6. Now we present an example for a large matrix. Consider $\Gamma_0(35)$. Let χ_1 be the primitive Dirichlet character with conductor $q_1 = 5$ such that $\chi_1(2) = -i$, and let χ_2 be the primitive Dirichlet character with conductor $q_2 = 7$ such that $\chi_2(3) = \exp(2\pi i(1/3))$. Let $\gamma = \begin{pmatrix} 46741638 & 43234369 \\ 43234205 & 39990117 \end{pmatrix}$. Computing $S_{\chi_1, \chi_2}(\gamma)$ by Definition 1.1 takes $5.531 * 10^4$ seconds (around 15 hours), whereas it takes $5.128 * 10^{-2}$ seconds using our algorithm.

3.4 Code

The algorithm discussed has been implemented using `Sage`. The reader can find the code at <https://github.com/prestontranbarger/NFDSFastComputation>.

Acknowledgements

This research was conducted at the 2022 REU hosted at Texas A&M University and supported by the National Science Foundation (DMS-2150094). The authors would like to thank Dr. Matthew Young for his continued support and input throughout the duration of the REU. The authors would like to thank Agniva Dasgupta for his immense help and feedback. The authors would also like to thank Mitch Majure for his input and contributions.

References

- [BR15] Matthias Beck and Sinai Robins. *Computing the continuous discretely*. Undergraduate Texts in Mathematics. Springer, New York, second edition, 2015.
- [DG20] Travis Dillon and Stephanie Gaston. An average of generalized Dedekind sums. *J. Number Theory*, 212:323–338, 2020.
- [Iwa97] H. Iwaniec. *Topics in Classical Automorphic Forms*. Graduate studies in mathematics. American Mathematical Soc., 1997.
- [Knu97] Donald E. Knuth. *The Art of Computer Programming Volume II: Seminumerical Algorithms*. Addison-Wesley Longman Publishing Co., Inc., USA, 1997.
- [LBVY21] Alexis LaBelle, Emily Van Bergeyk, and Matthew P. Young. Reciprocity and the kernel of Dedekind sums. *arXiv: 2110.12269*, 2021.
- [Maj22] Mitch Majure. Algebraic properties of the values of newform Dedekind sums. *arXiv: 2208.13060*, 2022.
- [MKS04] Wilhelm Magnus, Abraham Karrass, and Donald Solitar. *Combinatorial group theory*. Dover Publications, Inc., Mineola, NY, second edition, 2004.
- [NRY21] Evuilynn Nguyen, Juan J. Ramirez, and Matthew P. Young. The kernel of newform Dedekind sums. *J. Number Theory*, 223:53–63, 2021.
- [RG72] Hans Rademacher and Emil Grosswald. *Dedekind sums*. The Carus Mathematical Monographs, No. 16. Mathematical Association of America, Washington, D.C., 1972.
- [Ste07] William Stein. *Modular forms, a computational approach*, volume 79 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2007.
- [SVY20] T. Stucker, A. Vennos, and M. P. Young. Dedekind sums arising from newform Eisenstein series. *Int. J. Number Theory*, 16(10):2129–2139, 2020.