RISC

RESEARCH INSTITUTE FOR SYMBOLIC COMPUTATION

JKU

JOHANNES KEPLER UNIVERSITY LINZ

Computer-assisted construction of Ramanujan-Sato series for 1 over pi

Ralf Hemmecke, Peter Paule, Cristian-Silviu Radu

January 2025

RISC Report Series No. 25-01

ISSN: 2791-4267 (online)

Available at https://doi.org/10.35011/risc.25-01

JOHANNES KEPLER UNIVERSITY LINZ
Altenberger Str. 69
4040 Linz, Austria
www.jku.at
DVR 0093696

COMPUTER-ASSISTED CONSTRUCTION OF RAMANUJAN-SATO SERIES FOR 1 OVER π

RALF HEMMECKE, PETER PAULE, AND CRISTIAN-SILVIU RADU

Dedicated to George Andrews and Bruce Berndt on the occasion of passing milestone 85.

Abstract. Referring to ideas of Takeshi Sato, Yifan Yang in [30] described a construction of series for 1 over π starting with a pair (g, h), where g is a modular form of weight 2 and h is a modular function; i.e., a modular form of weight zero. In this article we present an algorithmic version, called "Sato construction". Series for $1/\pi$ obtained this way will be called "Ramanujan-Sato" series. Famous series fit into this definition, for instance, Ramanujan's series used by Gosper and the series used by the Chudnovsky brothers for computing millions of digits of π. We show that these series are induced by members of infinite families of Sato triples (N, γ_N, τ_N) where $N > 1$ is an integer and γ_N a 2×2 matrix satisfying $\gamma_N \tau_N = N\tau_N$ for τ_N being an element from the upper half of the complex plane.

In addition to procedures for guessing and proving from the holonomic toolbox together with the algorithm "ModFormDE", as described in [23], a central role is played by the algorithm "MultiSamba", an extension of Samba ("subalgebra module basis algorithm") originating from [24] and [16]. With the help of MultiSamba one can find and prove evaluations of modular functions, at imaginary quadratic points, in terms of nested algebraic expressions. As a consequence, all the series for $1/\pi$ constructed with the help of MultiSamba are proven completely in a rigorous non-numerical manner.

1. Introduction

In his famous early paper [25], Ramanujan presented seventeen series as different representations for 1 over π. Laying dormant for more than seventy years, this part of Ramanujan's work was brought back to the attention of a wider mathematical (and non-mathematical, e.g., [1] or [5]) audience by Bill Gosper and, around the same time, by Jon and Peter Borwein [4].

Date: January 31, 2025.

2010 Mathematics Subject Classification: primary 05A30, 11F03, 68W30; secondary 11F33, 11P83.

Keywords and phrases: modular forms and functions, holonomic differential equations, Ramanujan-Sato series for 1 over π, MultiSamba algorithm.

Both Gosper and the Borweins were after efficient methods to compute digits of π: the Borweins by generalizing methods tracing back to Gauß' arithmetic-geometric mean (AGM) algorithm, Gosper by using the last series listed by Ramanujan [25, id. (44)] which is,

$$\frac{1}{\pi} = \frac{\sqrt{8}}{9801} \cdot \sum_{n=0}^{\infty} (26390n + 1103) \frac{(4n)!}{n!^4} \left(\frac{1}{396^4}\right)^n. \tag{1}$$

Based on this series, Gosper computed a continued fraction approximation to π (his primary project goal) which converted to a decimal expansion gave more than 17 million digits of π, a world record at that time. Comparing this to the beginning of the 18th century when about 100 digits of π were known, Askey commented [1, p. 895], "Part of the improvement comes from larger and faster machines, but much more comes from increased mathematical knowledge. Part of this is specific to certain numbers, and part is general and of wide applicability. All of it is very interesting."

Indeed, the accomplishments of Gosper and the Borweins stimulated an avalanche of related developments; see, for example, the survey [2] by Baruah, Berndt, and Chan, the work by Chan and Cooper [7] presenting 186 series for $1/\pi$, Cooper's beautiful and comprehensive monograph [13], or Zudilin's article [31] entitled "Ramanujan-type formulae for $1/\pi$: A second wind?"

With the present article it is our hope to bring in new "algorithmic wind" into this area. One ingredient is based on holonomic methods: "ModFormDE", an algorithm for proving conjectured linear differential equations satisfied by modular forms. The coefficients of such differential equations are polynomials in a given modular function. Details are given in [22] and [23].

The second ingredient, described in this article, is an algorithmic construction of Ramanujan-Sato series in which the algorithm "MultiSamba" is used as an essential tool. With the help of MultiSamba one can find and prove evaluations of modular functions, at imaginary quadratic points, in terms of nested algebraic expressions.

Some words on the mathematical background of Ramanujan-Sato series are in place. As an application of his work on differential equations for modular forms, Yifan Yang [30] referred to a construction of series for 1 over π used by Takeshi Sato. Apart from the abstract [26], to our knowledge there is no further publication by Sato providing more details.

In the literature one finds various notions of "Ramanujan-Sato" series. In this article, the term *Ramanujan-Sato series* is reserved for a series of the type as on the right side of (13) which, in addition, can be obtained via the Sato construction as described in Section 3.

Remark 1.1. According to Heng Huat Chan [8], the notion "Ramanujan-Sato type series" appeared the first time in the title of [9]. The series for 1 over π presented there were motivated by series discovered by Sato, but were derived differently from the Sato construction.

The centerpiece of this article is the detailed algorithmic specification of the Sato construction in Section 3. We present a variety of examples to demonstrate the flexibility of our method. Concerning the series for $1/\pi$ which were derived using MultiSamba, we want to stress the following point: all the constants involved, i.e., the singular values of modular functions which find representations in terms of nested radicals, are fully proven.

The structure of this article is as follows.

Section 2 introduces all the basic notions used.

Section 3 presents Steps 0 to 8 of our algorithmization of the Sato construction described by Yang. All the steps are illustrated with the successive assembly of Ramanujan's series (1) as a concrete running example. As already mentioned, the derivation using MultiSamba gives a complete proof of (1).

Section 4 presents an infinite family of Sato triples. Each such triple, in combination with the modular form setting used to obtain (1), gives a series for $1/\pi$ as the result of the Sato construction. Ramanujan's (1) is a member of this family. Another family member is $\phi(m)$, a sum (85) which in the limit $m \to \infty$ gives $1/\pi$ and which adds 16 correct digits with each additional summand. This infinite Sato family can be viewed as an algorithmic counterpart to a general theorem by the Borweins [4, eq. (5.5.16)]; see Theorem 4.4 below. Hemmecke's implementation of our method in the computer algebra system FriCAS [15] derives the series (1) and other members from the infinite "Ramanujan-Gosper family" at the push of a button.

Section 5 presents two infinite families of Sato triples which induce series for $1/\pi$ as the result of the Sato construction. A prominent member of family 1 is the Chudnovsky series (87) which adds 14 correct digits with each additional summand, and which has been used until today for world-record computations of the digits of π; see [29]. The construction of series for $1/\pi$ from these families can be viewed as an algorithmic counterpart to a general theorem by the Chudnovskys [11, eq. (1.4)], Theorem 5.4 below, which in its original formulation covers family 1 explicitly. Using Hemmecke's implementation of MultiSamba in FriCAS derives the series (87) and other members from the infinite "Chudnovsky families 1 and 2" at the push of a button.

Section 6 presents again two infinite families of Sato triples inducing series for $1/\pi$. These series in a more direct way relate to the original work of Sato or, more precisely, to the work of Heng Huat Chan and collaborators [9] inspired by Sato. The underlying modular form setting is borrowed from work of Frits Beukers in connection with Apéry numbers. The construction of series for $1/\pi$ from the first Apéry-Beukers-Chan family can be viewed as an algorithmic counterpart to a general theorem by Heng Huat Chan and Helen Verrill [10, eq. (5.1)]; see Theorem (6.3) below.

Section 7 is devoted to a description of our main computational engine, the algorithm MultiSamba to derive and prove algebraic relations between modular functions. When explaining the functionality of the algorithm MultiSamba, which has a sophisticated polynomial reduction procedure as its main ingredient, we restrict to the presentation of illustrating examples.

Section 8 and Section 9 deal with those tasks of the algorithmic Sato construction for which MultiSamba is responsible. Step 0 of the construction is the choice of a pair (g, h), where g is a modular form of weight 2 and h is a modular function; i.e., a modular form of weight zero. The series ingredient of the Sato construction is a local expansion of the form,

$$g(\tau) = \sum_{n\geq 0} c(n)h(\tau)^n \quad \text{for all } \tau \in \mathbb{H} \text{ with } \Im(\tau) \text{ sufficiently large.} \tag{2}$$

The evaluation of $t_N := h(\tau_N)$ with τ_N an imaginary quadratic point is of special interest. Section 8 describes how an algebraic expression for t_N, ideally in the form of nested radicals, can be derived and proven algorithmically. Section 9 is devoted to a similar task of obtaining algebraic expressions for $p_1(t_N)$ and $p_2(t_N)$ where the p_j are algebraic functions defined using g.

Sections 10 and Section 11 deal with algorithmic aspects related to (2); explanations as a concrete example use the pair (g, h) which makes the starting point for the construction of (1). Section 10 describes the algorithmic discovery and proving of local expansions (2) using the holonomic toolbox as described in detail in [23]. Section 11 describes, again for this concrete case, how a bound $L > 0$ is derived such that for all τ with $\Im(\tau) > L$ the local expansion (2) holds.

In Section 12, for the sake of completeness, we prove the fact stated in Lemma 3.14; namely, that the modularity of the modular forms g and h, used in the construction of (1), indeed extends from $\Gamma(2)$ to a bigger group.

We conclude this Introduction with a remark on software: the RISC packages used, Mallinger's package `GeneratingFunctions` written in Mathematica and Hemmecke's package `QEta` written in FriCAS, are freely available at `https://caa.risc.jku.at/software`.

2. Basic notions

In this section we introduce basic notions used throughout the paper.

For certain hypergeometric series arising in the text some readers could prefer standard ${}_pF_q$-notation. To facilitate translation we use rising factorials,

$$(a)_n := a(a+1)\dots(a+n-1), n \geq 1, \text{ and } (a)_0 := 1.$$

By τ we denote an element of the upper half of the complex plane; i.e., $\tau \in \mathbb{H} = \{\tau \in \mathbb{C} : \Im(\tau) > 0\}$. Moreover,

$$q := \exp(2\pi i \tau), \text{ and } q_w := \exp(2\pi i \tau / w) \text{ for } w \in \mathbb{Z}_{\geq 1}.$$

It will be convenient to introduce an additional short hand for the general Fourier variable if w is clear from the context,

$$x := q_w = \exp(2\pi i \tau / w).$$

Throughout, Γ will be a congruence subgroup of $\mathrm{SL}_2(\mathbb{Z})$, or a subgroup of $\mathrm{GL}_2^+(\mathbb{Q})$ which in our context will be constructed as an extension of a congruence subgroup. The action of Γ on elements from $\mathbb{H} \cup \mathbb{Q} \cup \{\infty\}$ is as usual,

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \tau := \frac{a\tau + b}{c\tau + d}.$$

For any complex function f on $\mathbb{H}$ and $k \in \frac{1}{2}\mathbb{Z}$ the slash operator $|_k$ for $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma$ is defined by

$$(f \mid_k \gamma)(\tau) := \det(\gamma)^{k/2}(c\tau + d)^{-k} f(\gamma\tau).$$

To gain extra flexibility we allow for a character χ as a possible extra factor.

Definition 2.1. *Let f be a complex function on $\mathbb{H}$. We call f a modular form with weight $k \in \frac{1}{2}\mathbb{Z}$ and character χ, if it satisfies the following three conditions:*

(1) f is meromorphic on $\mathbb{H}$;
(2) $(f \mid_k \gamma)(\tau) = \chi(\gamma) f(\tau)$ for any $\gamma \in \Gamma$;
(3) f is meromorphic at each cusp point of Γ.

The set of all such modular forms is denoted by $M_k(\Gamma; \chi)$ which is a vector space over $\mathbb{C}$. If $\chi(\gamma) = 1$ for all $\gamma \in \Gamma$ we use the short hand,

$$M_k(\Gamma) := M_k(\Gamma, 1).$$

If, in addition, $k = 0$ the elements of $M_0(\Gamma)$ form the field of meromorphic modular functions for Γ.

For $f \in M_k(\Gamma; \chi)$ condition (3) of Definition 2.1 implies at the cusp ∞ a representation of the form

$$f(\tau) = \sum_{n \geq m} a_n \exp(2\pi i \tau / w)^n \quad \text{for all } \tau \text{ with } \Im(\tau) \text{ sufficiently large,}$$

and with $w \in \mathbb{Z}_{\geq 1}$ fixed. If $a_m \neq 0$ then $\mathrm{ord}_\infty(f) := m$ is the (vanishing) order of f at infinity. It will be convenient to define

$$\tilde{f}(z) := \sum_{n \geq m} a_n z^n.$$

This means, for all τ with $\Im(\tau)$ sufficiently large we have

$$f(\tau) = \tilde{f}(q_w). \tag{3}$$

Local expansions $(g, h, \Gamma; Y)$. Given $g(\tau) \in M_2(\Gamma; \chi)$ such that $\mathrm{ord}_\infty(g) \geq 0$, and $h(\tau) \in M_0(\Gamma)$ such that $\tilde{h}(z) = z + a_2 z^2 + a_3 z^3 + O(z^4)$. Then there exists a local expansion such that for all $\tau \in \mathbb{H}$ with $\Im(\tau)$ sufficiently large,

$$g(\tau) = Y(h(\tau)) \text{ where } Y(z) = \sum_{n=0}^{\infty} c(n) z^n \tag{4}$$

$$\text{with } (c(n))_{n \geq 0} \text{ being a holonomic sequence.}$$

It is a non-trivial fact that the $c(n)$ indeed form a *holonomic* sequence, see the discussion related to Proposition 6.2 in [23].

Definition 2.2. *A tuple* $(g, h, \Gamma; Y)$ *is called a local expansion if the series* Y *relates* $g \in M_2(\Gamma; \chi)$ *and* $h \in M_0(\Gamma)$ *as in* (4).

The Sato Eigenvalue Problem (Sato-EVP). Given Γ as a congruence subgroup of $\mathrm{SL}_2(\mathbb{Z})$, or as a subgroup of $\mathrm{GL}_2^+(\mathbb{Q})$ being an extension of a congruence subgroup; find $N \in \mathbb{Z}_{\geq 2}$, $\gamma_N = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma$, and $\tau_N \in \mathbb{H}$ such that

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \tau_N = N \tau_N. \tag{5}$$

Definition 2.3. *We call* $(N, \gamma_N = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right), \tau_N)$ *as in* (5) *a Sato triple for* Γ.

Example 2.4. One can easily verify,

$$(N, \gamma_N, \tau_N) = \left(233, \begin{pmatrix} -231 & 116 \\ -2 & 1 \end{pmatrix}, \frac{116 + i\sqrt{58}}{233}\right) \quad \text{is a Sato triple for } \Gamma := \Gamma(2).$$

Remark 2.5. If Γ is a classical congruence subgroup, there are straight-forward procedures to find Sato triples computationally. For example, if τ_N solves (5), then it must be a root of the polynomial

$$p(\tau) = Nc\tau^2 + N\tau d - a\tau - b; \tag{6}$$

i.e.,

$$\tau = \frac{-(Nd - a) \pm \sqrt{\Delta}}{2Nc} \tag{7}$$

where $\Delta = (Nd - a)^2 + 4Nbc = (Nd + a)^2 - 4N \det(\gamma_N)$ is the discriminant of $p(\tau)$. Since we want (non-real) solutions in $\mathbb{H}$, we restrict to $\Delta < 0$; i.e., $v^2 < 4N \det(\gamma)$ where $v := (dN + a) \in \mathbb{Z}$. For $\Gamma \leq \mathrm{SL}_2(\mathbb{Z})$ and fixed N, this gives only finitely many candidates for v. For each of these candidates and for each $d \in \mathbb{Z}$, we can find an $a := v - dN$ and corresponding matrix entries b and c to obtain a $\gamma_N = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ with $\det(\gamma_N) = 1$ such that (5).

In fact, the choice of c is under the condition that the expression for b yields an integer and $\gamma_N \in \Gamma$, the respective group under consideration; i.e., we have

$$\gamma_N = \begin{pmatrix} v - dN & \frac{d(v-dN)-1}{c} \\ c & d \end{pmatrix}. \tag{8}$$

Hence to satisfy these conditions, we only need to try values from suitably fixed (finite) domains for v, c, and d. Note that two solutions (γ_1, τ_1) and (γ_2, τ_2) with $\gamma_i \tau_i = N\tau_i$, $i \in \{1, 2\}$, can be considered as equivalent if τ_1 and τ_2 are equivalent; i.e., if there is a matrix $\gamma \in \Gamma$ with $\tau_2 = \gamma\tau_1$.

Example 2.6. Consider the following subgroup of $\mathrm{GL}_2^+(\mathbb{Q})$,

$$\Gamma := \left\langle \Gamma(2), \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & -2 \\ 1 & 0 \end{pmatrix} \right\rangle; \tag{9}$$

i.e., the group generated by the elements of $\Gamma(2)$ together with $\gamma_1 := \left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$ and $\gamma_2 := \left(\begin{smallmatrix} 0 & -2 \\ 1 & 0 \end{smallmatrix}\right)$. Note that $\det(\gamma_2) = 2$. One can easily verify:

$$(N, \gamma_N, \tau_N) = \left(29, \begin{pmatrix} 58 & -2 \\ 59 & -2 \end{pmatrix}, \frac{2}{59}\frac{58 + i\sqrt{58}}{58}\right) \quad \text{is a Sato triple for } \Gamma. \tag{10}$$

Subsequently we will use that

$$\begin{pmatrix} 58 & -2 \\ 59 & -2 \end{pmatrix} = \gamma_1\, \gamma_2\, {\gamma_1}^{-29} \in \Gamma. \tag{11}$$

Remark 2.7. In case that Γ is an extension of a classical congruence subgroup, to find Sato triples computationally one has to modify the procedure from the previous remark. If Γ is the group of Example 2.6, then one can show that

$$\Gamma = \{4^n\gamma : n \in \mathbb{Z} \text{ and } \gamma \in \Gamma_1 \cup \Gamma_2\},$$

where

$$\Gamma_1 = \left\{\gamma \in \mathrm{SL}_2(\mathbb{Z}) : \gamma \equiv \begin{pmatrix} 1 & 0 \\ * & 1 \end{pmatrix} \pmod 2\right\} \text{ and}$$

$$\Gamma_2 = \left\{\gamma \in \mathrm{GL}_2^+(\mathbb{Z}) : \det(\gamma) = 2, \gamma \equiv \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \pmod 2\right\}.$$

Thus, one basically has to run the search twice, first for Γ_1 as in Example 2.6 and then for Γ_2 where the 1 in (8) which corresponds to the determinant of γ_N is replaced by 2.

3. Algorithmization of the Sato construction described by Yang

In this section, the centerpiece of our article, a detailed algorithmic specification of the Sato construction is given. All the steps are illustrated with the incremental assembly of Ramanujan's series (1) as a concrete running example.

3.1. **General background of Sato's construction.** Despite using a slightly more general setting, the steps below follow closely the Sato construction given by Yang [30, pp. 3–4].

Nevertheless, we introduce a significant difference to Yang's description: *we present an algorithmic version for each of the steps.* This also motivates the following input/output specification of the procecure which we call Sato construction:

Input. A local expansion $(g, h, \Gamma; Y)$ with $g \in M_2(\Gamma; \chi)$ and $h \in M_0(\Gamma)$ as in Definition 2.2; i.e., for all $\tau \in \mathbb{H}$ with $\Im(\tau)$ sufficiently large,

$$g(\tau) = Y(h(\tau)) \text{ where } Y(z) = \sum_{n=0}^{\infty} c(n) z^n \tag{12}$$

for some holonomic sequence $c(n)$. In addition, we require that we are able to compute sufficiently many coefficients of the series representations $\tilde{g}(x)$ and $\tilde{h}(x)$.

Output. Integers A, B, an algebraic number $C \in \mathbb{Q}[\zeta]$, and an algebraic number $t \in \mathbb{Q}[\xi]$ such that

$$\frac{1}{\pi} = C \cdot \sum_{n=0}^{\infty} (An + B) c(n) t^n. \tag{13}$$

In the literature one finds various notions of "Ramanujan-Sato" series. In this article, the term *Ramanujan-Sato series* is reserved for series of the type as on the right side of (13) which, as an additional requirement, can be obtained by the Sato construction.

Example 3.1. We will show that Ramanujan's series (1), which can be rewritten as

$$\frac{1}{\pi} = \frac{2\sqrt{2}}{9801} \cdot \sum_{n=0}^{\infty} (26390n + 1103) \frac{(1/4)_n (1/2)_n (3/4)_n}{(1)_n (1)_n n!} 256^n \left(\frac{1}{396^4}\right)^n, \tag{14}$$

is indeed a Ramanujan-Sato series. To start the algorithmic construction of (14), we take as input $(g, h, Y; \Gamma)$ where

$$\Gamma := \Gamma(2), \tag{15}$$

$$g(\tau) := (1 + \lambda(\tau))\theta_3(\tau)^4 \in M_2(\Gamma), \tag{16}$$

$$h(\tau) := \frac{\lambda(\tau)(1 - \lambda(\tau))^2}{16(1 + \lambda(\tau))^4} \in M_0(\Gamma), \tag{17}$$

and

$$Y(z) := \sum_{n=0}^{\infty} \underbrace{\frac{(1/4)_n(1/2)_n(3/4)_n}{(1)_n(1)_n n!} 256^n}_{=c(n)} \cdot z^n. \tag{18}$$

The corresponding relation $g(\tau) = Y(h(\tau))$ is a rewritten version of the Borwein relation [4, Thm. 5.7b(iv)]; for our rewriting we use the Jacobi theta function $\theta_3(\tau) := \theta_3(0, \tau)$ and the modular lambda function $\lambda(\tau)$. For definitions and properties of these functions we refer to the open source library FunGrim [18] which we found very useful for the applications under discussion. For example, FunGrim at one place provides all the function properties needed to verify the memberships $g \in M_2(\Gamma(2))$ and $h \in M_0(\Gamma(2))$.

Particularly relevant to the algorithmic construction of Ramanujan-Sato series is the fact that given (g, h, Γ), the holonomic coefficient sequence $c(n)$ of $Y(z)$ can be determined also in an algorithmic fashion; see Section 10.

Remark 3.2. The prominence of (1), resp. (14), is owing to the fact that in 1985 Bill Gosper was successful in computing more than 17 million digits of π using (14). We present a quote from Richard Askey's insightful and also entertaining review [1] of the book [4] by Jon and Peter Borwein: "Gosper asked if I knew how to prove (14), and I had to admit I did not. Ramanujan had not given a proof. [...] If you are curious about how to prove (14), an outline is given in Chapter 5 of [4]."

Applying Step 1 to Step 8 of the Sato construction to our running example, we will derive *and* prove Ramanujan's series (14) used by Gosper.

3.2. **Step 0. Start with a local expansion.** The starting point of the Sato construction is an input in the form of a local expansion $(g, h, \Gamma; Y)$ as in (12) where we added the requirement that we are able to compute sufficiently many coefficients in the series representations,

$$\tilde{g}(x) := \sum_{n\geq 0} g_n x^n \ \ and \ \ \tilde{h}(x) := x + \sum_{n\geq 2} h_n x^n,$$

where

$$x = q_w = \exp(2\pi i\tau/w).$$

Recall that the equalities,

(19)

$$g(\tau) = \sum_{n\geq 0} g_n \exp(2\pi i\tau/w)^n (= \tilde{g}(x)) \ \textit{ and } \ h(\tau) = \sum_{n\geq 0} h_n \exp(2\pi i\tau/w)^n (= \tilde{h}(x)),$$

in general hold only when $\Im(\tau)$ is sufficiently large.

Example 3.3 (Ex. 3.1 contd.). We proceed with the input data from Example 3.1 to illustrate the steps of the construction. At this point, we set up the corresponding x-series. We have $w = 2$; i.e.,

$$x = q_2 = \exp(\pi i\tau).$$

With this setting one has the classical product representation for λ; e.g. [18]:

$$\lambda(\tau) = \frac{\theta_2(\tau)^4}{\theta_3(\tau)^4} \in M_0(\Gamma(2)) \tag{20}$$

$$= 16x \left[\frac{\prod_{n=1}^{\infty}(1-x^{2n})(1+x^{2n})^2}{\prod_{n=1}^{\infty}(1-x^{2n})(1+x^{2n-1})^2} \right]^4$$

$$= 16x - 128x^2 + 704x^3 - 3072x^4 + O(x^5) = \tilde{\lambda}(x).$$

This gives the required x-series representation for $h(\tau) = \frac{\lambda(\tau)(1-\lambda(\tau))^2}{16(1+\lambda(\tau))^4}$,

$$\tilde{h}(x) = x - 104x^2 + 6444x^3 - 311744x^4 + 13018830x^5 + O(x^6) \tag{21}$$

and similarly also the one for $g(\tau) = (1 + \lambda(\tau))\theta_3(\tau)^4$,

$$\tilde{g}(x) = 1 + 24x + 24x^2 + 96x^3 + 24x^4 + 144x^5 + O(x^6). \tag{22}$$

The x-series representations of g and h are taken as input to compute the local expansion $(g, h, \Gamma; Y)$.

For example, with the x-series (21) and (22) in hand, one determines that

$$c(n) = \frac{(1/4)_n(1/2)_n(3/4)_n}{(1)_n(1)_n n!} 256^n = \frac{(4n)!}{(n!)^4} \tag{23}$$

such that

$$g(\tau) = \sum_{n=0}^{\infty} c(n)h(\tau)^n \ \text{ for } \ \Im(\tau) \ \text{ sufficiently large;} \tag{24}$$

i.e., $Y(z)$ is as in (18). How this is done algorithmically is described in Section 10.

Now we are ready to proceed with the actual steps of the Sato construction.

3.3. **Step 1. Implement the Sato function G.** Define

$$G(\tau) := \frac{w}{2\pi i}\,\frac{g'(\tau)}{g(\tau)} = x\,\frac{\tilde{g}'(x)}{\tilde{g}(x)} = \tilde{G}(x). \tag{25}$$

The function G is not a modular but a quasi-modular form:

Lemma 3.4. *For all* $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma$,

$$G(\gamma\tau) = \frac{wc}{\pi i \cdot \det(\gamma)}\,(c\tau+d) + \frac{1}{\det(\gamma)}(c\tau+d)^2 G(\tau). \tag{26}$$

Proof. Obviously,

$$\frac{d}{d\tau} g(\gamma\tau) = g'(\gamma\tau)\frac{d}{d\tau}(\gamma\tau) = g'(\gamma\tau)\,\frac{\det(\gamma)}{(c\tau+d)^2}. \tag{27}$$

Owing to $g \in M_2(\Gamma;\chi)$, the left hand side equals

$$\begin{aligned} \frac{d}{d\tau} g(\gamma\tau) &= \frac{d}{d\tau}\frac{\chi(\gamma)}{\det(\gamma)}(c\tau+d)^2 g(\tau) \\ &= \frac{\chi(\gamma)}{\det(\gamma)}\,\big(2c(c\tau+d)g(\tau) + (c\tau+d)^2 g'(\tau)\big)\,. \end{aligned}$$

This implies,

$$\begin{aligned} \frac{2\pi i}{w}\,G(\gamma\tau) = \frac{g'(\gamma\tau)}{g(\gamma\tau)} &= \frac{(c\tau+d)^2}{g(\gamma\tau)}\frac{\chi(\gamma)}{\det(\gamma)^2}\,\big(2c(c\tau+d)g(\tau) + (c\tau+d)^2 g'(\tau)\big) \\ &= \frac{1}{\det(\gamma)}\left(2c(c\tau+d) + (c\tau+d)^2\frac{g'(\tau)}{g(\tau)}\right), \end{aligned}$$

which proves the statement. □

Example 3.5 (Ex. 3.1 contd.). The series expansion of $\tilde{g}(x)$ from (22) gives,

$$\tilde{G}(x) = x\,\frac{\frac{d}{dx}\big(1+24x+24x^2+96x^3+O(x^4)\big)}{1+24x+24x^2+96x^3+O(x^4)} = 24x - 528x^2 + O(x^3). \tag{28}$$

3.4. **Step 2. Implement the Sato function H.** The definition of the Sato H function is with respect to a fixed integer $N \geq 2$.

Definition 3.6. *Define*

$$H(\tau) := G(\tau) - N\,G(N\tau) = \tilde{G}(x) - N\,\tilde{G}(x^N) = \tilde{H}(x) \tag{29}$$

Lemma 3.7. *Let*

$$\Gamma' := \left\{ \begin{pmatrix} a & b \\ cN & d \end{pmatrix} \in \Gamma : \begin{pmatrix} a & bN \\ c & d \end{pmatrix} \in \Gamma \right\} \leq \Gamma. \tag{30}$$

Then for all $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma'$,

$$H\left(\begin{pmatrix} a & b \\ c & d \end{pmatrix}\tau\right) = \frac{1}{ad-bc}(c\tau+d)^2 H(\tau). \tag{31}$$

This means, $H \in M_2(\Gamma')$ *is a modular form of weight* 2 *with trivial character.*

Proof. For $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma'$ one has

$$\begin{aligned} G(N\gamma\tau) &= G\left(\frac{a(N\tau)+Nb}{\frac{c}{N}(N\tau)+d}\right) = G\left(\begin{pmatrix} a & bN \\ \frac{c}{N} & d \end{pmatrix}(N\tau)\right) \\ &\overset{(26)}{=} \frac{w\frac{c}{N}}{\pi i \cdot \det(\gamma)}(c\tau+d) + \frac{1}{\det(\gamma)}(c\tau+d)^2 G(N\tau) \end{aligned}$$

Consequently, applying (26) again,

$$\begin{aligned} H(\gamma\tau) &= G(\gamma\tau) - N\,G(N\gamma\tau) \\ &= \frac{wc}{\pi i \cdot \det(\gamma)}(c\tau+d) + \frac{1}{\det(\gamma)}(c\tau+d)^2 G(\tau) \\ &\quad - N\frac{w\frac{c}{N}}{\pi i \cdot \det(\gamma)}(c\tau+d) - N\frac{1}{\det(\gamma)}(c\tau+d)^2 G(N\tau) \\ &= \frac{1}{\det(\gamma)}(c\tau+d)^2 \underbrace{(G(\tau) - NG(N\tau))}_{=H(\tau)}. \end{aligned}$$

This proves the statment. □

Example 3.8 (Ex. 3.1 contd.). With the choice $N = 29$ the series expansion of $\tilde{G}(x)$ from (28) gives,

$$\tilde{H}(x) = \tilde{G}(x) - 29\,\tilde{G}(x^{29}) = 24x - 528x^2 + 12384x^3 + O(x^4). \tag{32}$$

3.5. **Step 3. Find a Sato triple for** Γ**.** Suppose one finds a Sato triple (N, γ_N, τ_N) for Γ, then

$$G(\gamma_N\tau_N) = G(N\tau_N), \tag{33}$$

which allows for a crucial interplay between G and H.

Lemma 3.9 (Sato relation for $1/\pi$)**.** *Given a modular form* $g \in M_2(\Gamma;\chi)$ *and a modular function* $h \in M_0(\Gamma)$*. Let*

$$\left(N, \gamma_N = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \tau_N\right) \quad \textit{be a Sato triple for } \Gamma. \tag{34}$$

Then for the associated Sato functions G *and* H *defined in* (25) *and* (3.6)*, where the* N *from* (34) *is used to define* H*, one has*

$$\frac{1}{\pi} = \alpha\,G(\tau_N) + \beta\,H(\tau_N), \tag{35}$$

where

$$\beta = -\frac{i}{wcN}\cdot\frac{\det(\gamma_N)}{c\tau_N+d} \quad \textit{and} \quad \alpha = -\frac{i}{wc}\cdot(c\tau_N+d)-\beta. \tag{36}$$

Proof. We begin with a rewritten version of (26),

$$\begin{aligned} -\frac{wc}{\pi i}(c\tau_N+d) &= (c\tau_N+d)^2G(\tau_N)-\det(\gamma_N)\,G(\gamma_N\tau_N)\\ &\overset{(33)}{=} (c\tau_N+d)^2G(\tau_N)-\det(\gamma_N)\,G(N\tau_N)\\ &\overset{(29)}{=} \left((c\tau_N+d)^2-\frac{\det(\gamma_N)}{N}\right)G(\tau_N)+\frac{\det(\gamma_N)}{N}H(\tau_N). \end{aligned}$$

Rearranging terms gives (35) together with (36). □

Example 3.10 (Ex. 3.1 contd.). Recall from Example 2.4 the Sato triple

$$(N,\gamma_N,\tau_N) = \left(233, \begin{pmatrix}-231 & 116\\ -2 & 1\end{pmatrix}, \frac{116+i\sqrt{58}}{233}\right) \text{ for } \Gamma := \Gamma(2). \tag{37}$$

Using Mathematica and its built-in functions for modular $\lambda(\tau)$ and $\theta_3(z,\tau)$, we verify numerically the Sato relation (35) for $g(\tau) = (1+\lambda(\tau))\theta_3(\tau)^4 \in M_2(\Gamma(2))$ as in (16):

In[1]:= $\mathbf{g[z\_, t\_] := (1 + ModularLambda[t]) * EllipticTheta[3, z, Exp[Pi\,I\,t]]^4}$

In[2]:= $\mathbf{tN = \frac{116 + i\sqrt{58}}{233};}$

In[3]:= $\mathbf{beta = -\frac{I}{2\cdot c\cdot 233}\cdot\frac{1}{c\,tN + d}\quad /.\{c \to -2, d \to 1\}//ComplexExpand}$

Out[3]= $-\frac{\sqrt{\frac{29}{2}}}{233} + \frac{i}{932}$

In[4]:= $\mathbf{alpha = -\frac{I}{2\cdot c}\cdot(c\,tN + d) - beta\quad /.\{c \to -2, d \to 1\}//ComplexExpand}$

Out[4]= $\frac{\sqrt{58}}{233}$

In[5]:= $\mathbf{G[t\_, z\_] = \frac{1}{Pi\,I}\frac{D[g[z, t], t]}{g[z, t]}}$

In[6]:= $\mathbf{N[(alpha + beta)G[z, t]/.\{t \to tN, z \to 0\}, 10] -}$
$\mathbf{N[233betaG[z, t]/.\{t \to 233tN, z \to 0\}, 10]}$

Out[6]= $0.3183098887 + 2.\times 10^{-10}i$

Comparison to $1/\pi$:

In[7]:= $\mathbf{N[1/Pi, 10]}$

Out[7]= 0.3183098862

Example 3.11 (Ex. 3.1 contd.). Recall from Example 2.6 the Sato triple

$$(N,\gamma_N,\tau_N)=\left(29,\begin{pmatrix}58&-2\\59&-2\end{pmatrix},\frac{2}{59}\frac{58+i\sqrt{58}}{58}\right) \tag{38}$$

$$\text{for }\Gamma:=\left\langle\Gamma(2),\begin{pmatrix}1&0\\1&1\end{pmatrix},\begin{pmatrix}0&-2\\1&0\end{pmatrix}\right\rangle=\langle\Gamma(2),\gamma_1,\gamma_2\rangle. \tag{39}$$

In Section 12 we show that

$$g(\tau)=(1+\lambda(\tau))\theta_3(\tau)^4\in M_2(\Gamma)\ \text{ for }\ \Gamma=\langle\Gamma(2),\gamma_1,\gamma_2\rangle; \tag{40}$$

concretely, for all $\gamma=\left(\begin{smallmatrix}a&b\\c&d\end{smallmatrix}\right)\in\langle\Gamma(2),\gamma_1,\gamma_2\rangle$,

$$g(\gamma\tau)=\chi(\gamma)\det(\gamma)^{-1}(c\tau+d)^2g(\tau), \tag{41}$$

where the character χ is determined uniquely by

$$\chi(\gamma_1)=1,\chi(\gamma_2)=-1,\ \text{ and }\ \chi(\gamma)=1\ \text{ for all }\gamma\in\Gamma(2).$$

Let us choose the function g as in (40). If we now verify numerically the Sato relation (35) for the Sato triple (38) by following the analogous Mathematica steps as in Example 3.10 and with the same precision of 10 digits, the Sato relation will be computed to the numerical value 0.318214511. This in comparison to the digits of $1/\pi$ is correct only for the first three places after the decimal point.

The explanation for this numerical discrepancy lies in the following facts:

$$\text{for }\ \tau_{29}=\frac{2}{59}\frac{58+i\sqrt{58}}{58}\ \text{ one has }\ \left|e^{\pi i\tau_{29}}\right|=0.986114\dots,$$

in contrast to the previous Sato value, where

$$\text{for }\ \tau_{233}=\frac{116+i\sqrt{58}}{233}\ \text{ one has }\ \left|e^{\pi i\tau_{233}}\right|=0.902411\dots,$$

which is still close but not that close to 1. In any case, this difference, looking small at the first glance, when using theta series expansions is sufficient to provide numerical problems for the evaluation of $G(\tau_{29})$.

A simple and classical way to overcome this numerical problem is to use the (quasi-)modular transformation property (26) for G. To this end, one takes an element $\tau_\infty\in\mathbb{H}$ which is closer to infinity (i.e., with larger imaginary part than τ_{29}) which maps via an element of Γ to τ_{29}. It is convient to make such a choice as follows,

$$\tau_{29}=\underbrace{\begin{pmatrix}0&-2\\1&-58\end{pmatrix}}_{=\gamma_1^{29}\gamma_2\in\Gamma}\tau_\infty\ \text{ where }\ \tau_\infty:=i\sqrt{58}. \tag{42}$$

Note that $|e^{\pi i \tau_\infty}| = 4.06649\cdots \times 10^{-11}$. By (26),

$$G(\tau_{29}) = G\left(\begin{pmatrix} 0 & -2 \\ 1 & -58 \end{pmatrix} \tau_\infty\right) = \frac{\tau_\infty - 58}{\pi i} + \frac{(\tau_\infty - 58)^2}{2} G(\tau_\infty). \tag{43}$$

Moreover, by (38),

$$G(29\,\tau_{29}) = G\left(\begin{pmatrix} 58 & -2 \\ 59 & -2 \end{pmatrix} \tau_{29}\right) = 59\frac{59\tau_{29} - 2}{\pi i} + \frac{(59\tau_{29} - 2)^2}{2} G(\tau_{29}). \tag{44}$$

This puts us into the position to obtain numerical evaluations with high precision.

```
In[8]:=  g[z_, t_] := (1 + ModularLambda[t]) * EllipticTheta[3, z, Exp[Pi I t]]^4
In[9]:=  G[z_, t_] = 1/(Pi I) D[g[z, t], t]/g[z, t]
In[10]:= {t29, tInf} = {2/59 (58 + I Sqrt[58])/58, I Sqrt[58]};
In[11]:= GtInf = N[G[z, t] /.{t -> tInf, z -> 0}, 30]
Out[11]= 9.75956586709235809370526683996 × 10^-10
In[12]:= Gt29 = (tInf - 58)/(Pi I) + (tInf - 58)^2/2 GtInf
Out[12]= 2.42417748378535246169612927403042753 + 18.4619729675653512672362567501793355 04i
In[13]:= G29t29 = 59 (59 t29 - 2)/(Pi I) + (59 t29 - 2)^2/2 Gt29
Out[13]= 4.84835168542870382023223163497961677 75 - 0.636619757502253491973664025868252 9484i
In[14]:= beta = -I/(2 · c · 29) · 2/(c t29 + d)   /.{c -> 59, d -> -2}//ComplexExpand
Out[14]= -1/(59 Sqrt[58])
In[15]:= alpha = -I/(2 · c) · (c t29 + d) - beta   /.{c -> 59, d -> -2}//ComplexExpand
Out[15]= Sqrt[2/29]/59
In[16]:= (alpha + beta)Gt29 - 29 betaG29t29
Out[16]= 0.318309886183790671537767526745028724 07 + 0. × 10^-39 i
```

Comparison to $1/\pi$:

```
In[17]:= N[1/Pi, 40]
Out[17]= 0.3183098861837906715377675267450287240 69
```

3.6. **Step 4. Compute the Sato constants α and β.** Given a Sato triple for Γ as in (34),

$$\left(N, \gamma_N = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \tau_N\right),$$

by Lemma 3.9 the Sato constants are determined as in (36),

$$\beta = -\frac{i}{wcN} \cdot \frac{\det(\gamma_N)}{c\tau_N + d} \quad \text{and} \quad \alpha = -\frac{i}{wc} \cdot (c\tau_N + d) - \beta.$$

In Example 3.10 we considered the Sato triple for $\Gamma = \Gamma(2)$, and computed in `Out[3]` and `Out[4]` the corresponding values of β and α, respectively.

Example 3.12 (Ex. 3.1 contd.). Example 3.11 is relevant for our running example. There we considered the Sato triple

$$(N, \gamma_N, \tau_N) = \left(29, \begin{pmatrix} 58 & -2 \\ 59 & -2 \end{pmatrix}, \frac{2}{59}\frac{58 + i\sqrt{58}}{58}\right) \tag{45}$$

$$\text{for } \Gamma := \left\langle \Gamma(2), \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & -2 \\ 1 & 0 \end{pmatrix} \right\rangle = \langle \Gamma(2), \gamma_1, \gamma_2 \rangle, \tag{46}$$

and computed in `Out[14]` and `Out[15]`,

$$\beta = -\frac{1}{59\sqrt{58}} \text{ and } \alpha = \frac{\sqrt{\frac{2}{29}}}{59}. \tag{47}$$

The remaining steps of the construction deal with the conversion of the Sato relation (35) from Lemma 3.9 into a series representation for $1/\pi$, more precisely, with the conversion of $G(\tau_N)$ and $H(\tau_N)$ into series representations. Now local expansions $(g, h, \Gamma; Y)$ come into play. Being of crucial importance, we recall the fundamental relation (4) between $g \in M_2(\Gamma; \chi)$ and $h \in M_0(\Gamma)$:

$$g(\tau) = Y(h(\tau)) \text{ where } Y(z) = \sum_{n=0}^{\infty} c(n) z^n \tag{48}$$
$$\text{and where } (c(n))_{n \geq 0} \text{ is a holonomic sequence.}$$

In view of (48) a natural idea to represent $G(\tau_N)$ and $H(\tau_N)$ is to use the series $Y(h(\tau_N))$. So the value $h(\tau_N)$ will play an important role.

3.7. **Step 5. Compute a closed form representation of $h(\tau_N)$.** To make things concrete, we continue our runnning example.

Example 3.13 (Ex. 3.1 contd.). Throughout our running example we fix g and h as in (16) and (17); i.e.,

$$g(\tau) := (1 + \lambda(\tau))\theta_3(\tau)^4 \in M_2(\Gamma(2)), \tag{49}$$

$$h(\tau) := \frac{\lambda(1 - \lambda(\tau))^2}{16(1 + \lambda(\tau))^4} \in M_0(\Gamma(2)), \tag{50}$$

In previous examples we already worked with the extended group,

$$\Gamma := \left\langle \Gamma(2), \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & -2 \\ 1 & 0 \end{pmatrix} \right\rangle = \langle \Gamma(2), \gamma_1, \gamma_2 \rangle. \tag{51}$$

As stated explicitly in the following lemma, the modularity of g and h as in (49) and (50) extends to this bigger group, a fact which is proven in Section 12.

Lemma 3.14. *We have,*

$$g \in M_2(\Gamma;\chi) \ \text{and}\ h \in M_0(\Gamma)\ \text{where}\ \Gamma = \langle \Gamma(2), \gamma_1, \gamma_2 \rangle, \tag{52}$$

and where the character χ *is determined uniquely by*

$$\chi(\gamma_1) = 1, \chi(\gamma_2) = -1,\ \text{and}\ \chi(\gamma) = 1\ \text{for all}\ \gamma \in \Gamma(2).$$

Consider the Sato triple (45) for Γ as in (46). By Lemma 3.14 and (42) we have for $\gamma_1^{29}\gamma_2 = \left(\begin{smallmatrix} 0 & -2 \\ 1 & -58 \end{smallmatrix}\right) \in \Gamma$,

$$h(\tau_{29}) = h(\gamma_1^{29}\gamma_2\ i\sqrt{58}) = h(i\sqrt{58}). \tag{53}$$

Next we determine this value numerically using `ModularLambda`, the built-in function Mathematica provides for λ:

```
In[18]:= ML = ModularLambda;
In[19]:= h[t_] := (1 - ML[t])^2 ML[t] / (16(1 + ML[t])^4)
In[20]:= N[h[I√58], 10]
Out[20]= 4.066485764 × 10^-11
```

The output indicates that $h(i\sqrt{58})$ is a real number. In view of 10^{-11} we look at the reciprocal value:

```
In[21]:= N[1/h[I√58], 10]
Out[21]= 2.459125786 × 10^10
In[22]:= N[1/h[I√58], 30]
Out[22]= 2.45912578560000000000000000000 × 10^10
```

The numerical stability displayed by the last output line suggests that $h(i\sqrt{58})$ is a rational number.

Lemma 3.15. *For* h *as in* (50) *and* τ_{29} *as in* (45)*,*

$$h(\tau_{29}) = h(i\sqrt{58}) = \frac{1}{2459125786} = \frac{1}{396^4}. \tag{54}$$

The proof of this lemma together with remarks on related computational aspects is given in Section 8. Our running example will be continued in the next section, Step 6.

In general, closed form representations of $h(\tau_N)$ depend on the functions defining h, and on available τ_N from Sato triples from which suitable candidates are chosen. To this end, lists as presented at [20] or [21] are helpful.

If no further information on such τ_N is available, one, in a first step, can try to derive a plausible guess for $h(\tau_N)$ numerically. Then to prove a conjectured closed form representation of $h(\tau_N)$, one either uses classical methods or algorithmic tools. Concerning the latter, further remarks are given in Section 8; see, in particular, Section 8.1 for an algorithmic derivation and proof of (54) done with the help of the MultiSamba algorithm.

3.8. **Step 6. Compute a series representation of $H(\tau_N)$.** The verification of the following fact is straight-forward.

Lemma 3.16. *For $g \in M_2(\Gamma;\chi)$ and $H \in M_2(\Gamma')$ as in (29), with Γ' as in (30),*

$$\frac{H(\tau)}{g(\tau)} \in M_0(\Gamma'_\chi) \tag{55}$$

where

$$\Gamma'_\chi := \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma' : \chi(a,b,c,d) = 1 \right\}.$$

As a modular function with respect to Γ'_χ, the function H/g satisfies an algebraic relation with $h \in M_0(\Gamma)$; notice that $h \in M_0(\Gamma'_\chi)$ owing to $\Gamma'_\chi \leq \Gamma' \leq \Gamma$. Consequently, H/g can be considered as an algebraic function in h, which justifies the notation

$$p_1(h(\tau)) := \frac{H(\tau)}{g(\tau)}, \tag{56}$$

where p_1 denotes the corresponding algebraic function.

One essential ingredient to Step 6 is to invoke the local expansion as in (4); i.e., for $g(\tau) \in M_2(\Gamma;\chi)$ and $h(\tau) \in M_0(\Gamma)$,

$$g(t) = Y(h(t)) \text{ where } Y(z) = \sum_{n=0}^{\infty} c(n) z^n, \tag{57}$$

which in general holds only for such $t \in \mathbb{H}$ where $\Im(t)$ is sufficiently large. In order to invoke (57) for concrete evaluation points t, one needs to determine a concrete neighborhood of infinity; i.e., a lower bound $L > 0$ such that (57) holds for all $t \in \mathbb{H}$ with $\Im(t) > L$. As a concrete example, in Section 11 we derive and prove the bound $L = 1.87$ for the local expansion (24) of our running example.

It can happen that the τ_N from a Sato triple does not lie in a neighborhood U_L of infinity induced by such a lower bound L. In such cases one can still proceed with the Sato construction if a point t equivalent to τ_N exists in U_L. In other words, if there exists a transformation $T \in \Gamma$ which allows to increase the imaginary part of τ_N such that

$$\text{for } t := T\tau_N \in \mathbb{H} \text{ the local expansion (57) holds.} \tag{58}$$

Suppose we have such a t, then setting $S := T^{-1} = \left(\begin{smallmatrix} s_1 & s_2 \\ s_3 & s_4 \end{smallmatrix}\right)$,

$$\begin{aligned} H(\tau_N) &= \frac{H(\tau_N)}{g(\tau_N)} \cdot g(\tau_N) = p_1(h(\tau_N)) \cdot g(St) \\ &= p_1(h(\tau_N)) \cdot \chi(S) \det(S)^{-1}(s_3 t + s_4)^2 g(t) \\ &\overset{(58)}{=} p_1(h(\tau_N)) \cdot \chi(S) \det(S)^{-1}(s_3 t + s_4)^2 \sum_{n=0}^{\infty} c(n) h(t)^n \\ &= \frac{\chi(S)(s_3 T\tau_N + s_4)^2}{\det(S)} \cdot p_1(h(\tau_N)) \sum_{n=0}^{\infty} c(n) h(\tau_N)^n, \end{aligned}$$

where the last equality is by $h(t) = h(T\tau_N) = h(\tau_N)$.

The occurence of p_1 reveals also the second essential ingredient to Step 6:

determine a closed form for $p_1(h(\tau_N))$.

Using our toolbox, also this task, in principle, can be done algorithmically. A description of the corresponding procedure is given in Section 9.1.

Summary of Step 6: First, one has to accomplish the following task:

Determine a bound $L > 0$ such that for all $t \in \mathbb{H}$ with $\Im(t) > L$,

$$g(t) = Y(h(t)) \text{ where } Y(z) = \sum_{n=0}^{\infty} c(n) z^n. \tag{59}$$

Next, determine $S = T^{-1} = \left(\begin{smallmatrix} s_1 & s_2 \\ s_3 & s_4 \end{smallmatrix}\right)$ such that $\Im(T\tau_N) > L$, which implies that

$$H(\tau_N) = \frac{\chi(S)(s_3 T\tau_N + s_4)^2}{\det(S)} \cdot p_1(h(\tau_N)) \sum_{n=0}^{\infty} c(n) h(\tau_N)^n, \tag{60}$$

with $p_1(h(\tau_N))$ as in (56). Finally, determine a closed form for $p_1(h(\tau_N))$.

Example 3.17 (Ex. 3.1 contd.). We continue our running example; i.e., with g and h and the Sato triple $(29, \gamma_{29}, \tau_{29})$ as in (49), (50), and (45), respectively. Recall that for all $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma := \langle \Gamma(2), \gamma_1, \gamma_2 \rangle$,

$$g(\gamma\tau) = \chi(\gamma) \det(\gamma)^{-1}(c\tau + d)^2 g(\tau), \tag{61}$$

where the character χ is determined uniquely by

$$\chi(\gamma_1) = 1, \chi(\gamma_2) = -1, \text{ and } \chi(\gamma) = 1 \text{ for all } \gamma \in \Gamma(2). \tag{62}$$

In addition, recall (42),

$$\tau_{29} = \underbrace{\begin{pmatrix} 0 & -2 \\ 1 & -58 \end{pmatrix}}_{=\gamma_1^{29}\gamma_2 \in \Gamma} \tau_\infty \text{ where } \tau_\infty := i\sqrt{58}, \tag{63}$$

and (54),

$$h(\tau_{29}) = h(i\sqrt{58}) = \frac{1}{396^4}.$$

In Section 11 we proved $L = 1.87$ as a bound for the local expansion,

$$g(t) = \sum_{n=0}^{\infty} \frac{(1/4)_n(1/2)_n(3/4)_n}{(1)_n(1)_n n!} 256^n h(t)^n, \ \Im(t) > 1.87. \tag{64}$$

Observing that

$$\Im(\tau_{29}) = 0.0044\ldots \quad \text{and} \quad \Im(\tau_\infty) = 7.61577\dots > 1.87,$$

and in view of $\left(\begin{smallmatrix} 0 & -2 \\ 1 & -58 \end{smallmatrix}\right)^{-1} \tau_{29} = \tau_\infty$ we apply (60) with

$$T := \begin{pmatrix} 0 & -2 \\ 1 & -58 \end{pmatrix}^{-1} = \begin{pmatrix} -29 & 1 \\ -\frac{1}{2} & 0 \end{pmatrix} \text{ and } S := T^{-1} = \begin{pmatrix} 0 & -2 \\ 1 & -58 \end{pmatrix} = \gamma_1^{29}\gamma_2. \tag{65}$$

Substituting into (60) gives,

$$H(\tau_{29}) = p_1\left(\frac{1}{396^4}\right) \cdot (-1)\frac{1}{2}(i\sqrt{58} - 58)^2 \sum_{n=0}^{\infty} c(n) \left(\frac{1}{396^4}\right)^n. \tag{66}$$

Finally, we take the quotient with numerator

`Gt29 - 29 G29t29`,

the value representing $H(\tau_{29})$ in `In[16]`, and with denominator being a truncated version of the series to the right of (66) to conjecture a closed form for $p_1(h(\tau_{29})) = p_1(h(1/396^4))$. Recall that `tInf` $= i\sqrt{58}$ has been defined in `In[10]`:

In[23]:= $\mathbf{PH = Pochhammer;}$

In[24]:= $\mathbf{c[n\_] := \frac{PH[1/4,n]PH[1/2,n]PH[3/4,n]}{PH[1,n]PH[1,n]*n!}256^n}$

In[25]:= $\mathbf{R[u\_] := Sum[-\frac{1}{2}(tInf-58)^2c[n](1/396^4)^n, \{n,0,u\}]}$

Using `Gt29` and `G29t29` from the Mathematica session in Example 3.13, we obtain a numerical value for $p_1\left(\frac{1}{396^4}\right)$:

In[26]:= $\mathbf{Np1 = \frac{Gt29 - 29\,G29t29}{N[R[10],30]}}$

Out[26]= $0.0835922693640600775637888827631 + 0.\times 10^{-31}\,\mathtt{i}$

Finally, we use Mathematica to guess an algebraic representation of $p_1\left(\frac{1}{396^4}\right)$:

In[27]:= $\mathbf{RootApproximant[Re[Np1]]}$

Out[27]= $\frac{4412}{9801\sqrt{29}}$

As described in Section 9.1, using the algorithm MultiSamba one can not only find but also prove this evaluation which we state as a lemma.

Lemma 3.18. *Let g and h and the Sato triple $(29, \gamma_{29}, \tau_{29})$ be as in Example 3.17. Define the algebraic function p_1 as above by*

$$p_1(h(\tau)) := \frac{H(\tau)}{g(\tau)}.$$

Then

$$p_1(h(\tau_{29})) = p_1(h(\tau_\infty)) = p_1\left(\frac{1}{396^4}\right) = \frac{4412}{9801\sqrt{29}}. \tag{67}$$

3.9. **Step 7. Compute a series representation of $G(\tau_N)$.** Step 7 works very similarly to Step 6.

Lemma 3.19. *Let $(g, h, \Gamma; Y)$ be a local expansion as in Definition 2.2, and let G be the associated Sato function as in (25). Then for all $\tau \in \mathbb{H}$ such that $\Im(\tau)$ sufficiently large,*

$$G(\tau) = \frac{w}{2\pi i}\frac{h'(\tau)}{g(\tau)} \cdot Y'(h(\tau)). \tag{68}$$

Proof. The chain rule used on (4),

$$g'(\tau) = \frac{dY(h(\tau))}{d\tau} = Y'(h(\tau)) \cdot h'(\tau),$$

gives

$$G(\tau) = \frac{w}{2\pi i}\frac{g'(\tau)}{g(\tau)} = \frac{w}{2\pi i}\frac{h'(\tau)}{g(\tau)} \cdot Y'(h(\tau)).$$

□

Also the next lemma is straight-forward to prove.

Lemma 3.20. *Let*

$$\Gamma_\chi := \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma : \chi(a, b, c, d) = 1 \right\},$$

then

$$\frac{h'(\tau)}{g(\tau)} \in M_0(\Gamma_\chi).$$

Proof. For $\gamma = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma_\chi$,

$$\begin{aligned} \frac{h'(\gamma\tau)}{g(\gamma\tau)} &= \frac{(c\tau + d)^2}{\det(\gamma)} \frac{d}{d\tau} h(\gamma\tau) \cdot \frac{1}{g(\gamma\tau)} = \frac{(c\tau + d)^2}{\det(\gamma)} h'(\tau) \cdot \frac{1}{g(\gamma\tau)} \\ &= \frac{(c\tau + d)^2}{\det(\gamma)} h'(\tau) \cdot \frac{1}{\chi(\gamma)} \frac{\det(\gamma)}{(c\tau + d)^2} \frac{1}{g(\tau)} = \frac{h'(\tau)}{g(\tau)}. \end{aligned}$$

□

As a modular function with respect to Γ_χ, the function h'/g satisfies an algebraic relation with $h \in M_0(\Gamma)$; notice that $h \in M_0(\Gamma_\chi)$ owing to $\Gamma_\chi \leq \Gamma$. Consequently, h'/g can be considered as an algebraic function in h, which justifies the notation

$$p_2(h(\tau)) := \frac{w}{2\pi i}\,\frac{h'(\tau)}{g(\tau)}, \tag{69}$$

where p_2 denotes the corresponding algebraic function.

As in Step 6, for τ_N taken from a Sato triple (N, γ_N, τ_N) for Γ, let $T \in \Gamma$ be a transformation such that

for $t := T\tau_N \in \mathbb{H}$ the local expansion (57), respectively (4), holds.

For such a t the relation (68) holds; i.e.,

$$G(t) = p_2(h(t)) \cdot Y'(h(t)).$$

Then, setting $S := T^{-1} = \left(\begin{smallmatrix} s_1 & s_2 \\ s_3 & s_4 \end{smallmatrix}\right)$, by (26),

$$\begin{aligned} G(\tau_N) = G(St) &= \frac{ws_3}{\pi i \cdot \det(S)}\,(s_3t + s_4) + \frac{1}{\det(S)}(s_3t+s_4)^2 G(t) \\ &= \frac{ws_3(s_3T\tau_N + s_4)}{\pi i \cdot \det(S)} + \frac{(s_3T\tau_N + s_4)^2}{\det(S)}\,p_2(h(\tau_N)) \cdot Y'(h(\tau_N)). \end{aligned}$$

The occurence of p_2 gives rise to a task similar to Step 6:

determine a closed form for $p_2(h(\tau_N))$.

Using our toolbox, also this task, in principle, can be done algorithmically. A description of the corresponding procedure is given in Section 9.2.

Summary of Step 7: As in Step 6, one has to accomplish the following task first:

Determine a bound $L > 0$ such that for all $t \in \mathbb{H}$ with $\Im(t) > L$,

$$g(t) = Y(h(t)) \text{ where } Y(z) = \sum_{n=0}^{\infty} c(n) z^n. \tag{70}$$

Next, determine $S = T^{-1} = \left(\begin{smallmatrix} s_1 & s_2 \\ s_3 & s_4 \end{smallmatrix}\right)$ such that $\Im(T\tau_N) > L$, which implies that

$$\begin{aligned} G(\tau_N) = &\frac{ws_3(s_3T\tau_N + s_4)}{\pi i \cdot \det(S)} \\ &+ \frac{(s_3T\tau_N + s_4)^2}{\det(S)} \cdot \frac{p_2(h(\tau_N))}{h(\tau_N)} \sum_{n=0}^{\infty} n\,c(n) h(\tau_N)^n, \end{aligned} \tag{71}$$

with $p_2(h(\tau_N))$ as in (69). Finally, determine a closed form for $p_2(h(\tau_N))$.

Example 3.21 (Ex. 3.1 contd.). We continue our running example; i.e., with g and h, the Sato triple $(29, \gamma_{29}, \tau_{29})$, and $S := T^{-1} = \left(\begin{smallmatrix} 0 & -2 \\ 1 & -58 \end{smallmatrix}\right)$ as in Example 3.17. Recall from there that for $t := T\tau_{29} = \tau_\infty = i\sqrt{58}$ one has $\Im(t) > L = 1.87$ and thus

$$g(t) = Y(h(t)) = \sum_{n=0}^{\infty} \frac{(1/4)_n (1/2)_n (3/4)_n}{(1)_n (1)_n n!} 256^n h(t)^n.$$

Substituting into (71) gives,

$$G(\tau_{29}) = \frac{i\sqrt{58} - 58}{\pi i} + \frac{(i\sqrt{58} - 58)^2}{2} \cdot \frac{p_2(1/396^4)}{1/396^4} \sum_{n=0}^{\infty} n\, c(n) \left(\frac{1}{396^4}\right)^n. \tag{72}$$

As in Example 3.17 we use this relation to conjecture a closed form for

$$p_2(h(\tau_{29})) = p_2\left(h\left(\frac{1}{396^4}\right)\right).$$

To this end, we define a numerical approximation `Gt29` to $G(\tau_{29})$ as `In[11]`, and a truncated version of the series to the right of (72). Recall that `c[n]` has been defined in `In[24]`:

In[28]:= $\mathbf{g[z\_, t\_] := (1 + ModularLambda[t]) * EllipticTheta[3, z, Exp[Pi\, I\, t]]^4}$

In[29]:= $\mathbf{G[z\_, t\_] = \frac{1}{Pi\, I} \frac{D[g[z,t], t]}{g[z,t]}}$

In[30]:= $\mathbf{tInf = I\sqrt{58};}$

In[31]:= $\mathbf{GtInf = N[G[z,t]\ /. \{t \to tInf, z \to 0\}, 120]}$

Out[31]= 9.75956586709235809370526683995692667489532659668959102399177925536346
899296654507256137802252038545334301529245012463637 × 10^{-10}

In[32]:= $\mathbf{Gt29 = \frac{tInf - 58}{Pi\, I} + \frac{(tInf - 58)^2}{2} GtInf;}$

In[33]:= $\mathbf{r[u\_] := Sum[n\, c[n] (1/396^4)^n, \{n, 0, u\}]}$

In[34]:= $\mathbf{NtInf = N[I\, Sqrt[58], 120];}$

In[35]:= $\mathbf{N[\left(Gt29 - \frac{2(NtInf - 58)}{Pi\, I \cdot 2}\right) \cdot \frac{2}{(NtInf - 58)^2} \cdot \frac{1}{396^4}, 120] \cdot \frac{1}{N[r[30], 120]}}$

Out[35]= 4.06648574322890548453030727765541005112444761638669300303785526323591
054137052068091708966676816792164769012316521 × 10^{-11} + 0. × 10^{-125} i

In[36]:= $\mathbf{RootApproximant[\%]}$

Out[36]= $\frac{455\sqrt{29}}{60254729561664}$

As described in Section 9.2, using the algorithm "MultiSamba" one can not only find but also prove this evaluation which we state as a lemma.

Lemma 3.22. *Let g and h and the Sato triple $(29, \gamma_{29}, \tau_{29})$ be as in Example 3.17. Define the algebraic function p_2 as above by*

$$p_2(h(\tau)) := \frac{w}{2\pi i} \frac{h'(\tau)}{g(\tau)}.$$

Then

$$p_2(h(\tau_{29})) = p_2(h(\tau_\infty)) = p_2\left(\frac{1}{396^4}\right) = \frac{455\sqrt{29}}{60254729561664}. \tag{73}$$

3.10. **Step 8. Setting up the $1/\pi$ series.** After Steps 1 to 7 we are now ready for setting up the resulting series for 1 over π. For convenience we recall the setting for our algorithmic construction.

Given $h \in M_0(\Gamma)$ and $g \in M_2(\Gamma;\chi)$ such that for all $\gamma = \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right) \in \Gamma$,

$$g(\gamma\tau) = \chi(\gamma)\det(\gamma)^{-1}(C\tau + D)^2 g(\tau),$$

the starting point of the Sato construction is input in the form of a local expansion $(g, h, \Gamma; Y)$ as in (12). An additional algorithmic requirement is that we are able to compute sufficiently many coefficients of the series representations,

$$\tilde{g}(x) := \sum_{n\geq 0} g_n x^n \quad \textit{and} \quad \tilde{h}(x) := x + \sum_{n\geq 2} h_n x^n,$$

where

$$x = q_w = \exp(2\pi i\tau/w).$$

Recall that

$$g(\tau) = \sum_{n\geq 0} g_n \exp(2\pi i\tau/w)^n = \tilde{g}(x) \quad \textit{and} \quad h(\tau) = \sum_{n\geq 0} h_n \exp(2\pi i\tau/w)^n = \tilde{h}(x)$$

in general hold only when $\Im(\tau)$ is sufficiently large.

Moreover, suppose that $(N, \gamma_N = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right), \tau_N)$ is a Sato triple for Γ; i.e., its entries satisfy the Sato EVP (5). In addition, let $S = T^{-1} = \left(\begin{smallmatrix} s_1 & s_2 \\ s_3 & s_4 \end{smallmatrix}\right)$ be such that for $t := T\tau_N$ the local expansion

$$g(t) = Y(h(t)) \text{ with } Y(z) = \sum_{n=0}^{\infty} c(n) z^n$$

holds. Let

$$\beta = -\frac{i}{wcN} \cdot \frac{\det(\gamma_N)}{c\tau_N + d} \quad \text{and} \quad \alpha = -\frac{i}{wc} \cdot (c\tau_N + d) - \beta.$$

be as in (36), and p_1 and p_2 as in (56) and (69), respectively, then the desired series representation for 1 over π is

$$\frac{1}{\pi} = \alpha\Big(\frac{ws_3(s_3T\tau_N + s_4)}{\pi i \cdot \det(S)} + \frac{(s_3T\tau_N + s_4)^2}{\det(S)} \cdot \frac{p_2(h(\tau_N))}{h(\tau_N)} \sum_{n=0}^{\infty} n\, c(n) h(\tau_N)^n\Big)$$

$$+ \beta\Big(\frac{\chi(S)(s_3T\tau_N + s_4)^2}{\det(S)} \cdot p_1(h(\tau_N)) \sum_{n=0}^{\infty} c(n) h(\tau_N)^n\Big). \tag{74}$$

This relation is simply obtained from the Sato relation (35) by filling in (71) for $G(\tau_N)$ and (60) for $H(\tau_N)$.

Example 3.23 (Ex. 3.1 contd.). Now we are ready to complete our running example where

$$x = q_2 = \exp(\pi i \tau).$$

Given the input data as in Example 3.21, we need to substitute the following items into (74):

$$\beta = -\frac{1}{59\sqrt{58}} \text{ and } \alpha = \frac{\sqrt{\frac{2}{29}}}{59} \qquad \text{(by (47))},$$

$$S = T^{-1} = \begin{pmatrix} s_1 & s_2 \\ s_3 & s_4 \end{pmatrix} = \begin{pmatrix} 0 & -2 \\ 1 & -58 \end{pmatrix} = \gamma_1^{29}\gamma_2 \qquad \text{(by (65))} ,$$

$$\chi(\gamma_1) = 1, \chi(\gamma_2) = -1, \text{ and } \chi(\gamma) = 1 \text{ for all } \gamma \in \Gamma(2) \qquad \text{(by (62))},$$

$$T\tau_{29} = \tau_\infty = i\sqrt{58} \qquad \text{(by (63))},$$

$$h(\tau_{29}) = h(i\sqrt{58}) = \frac{1}{396^4} \qquad \text{(by (54))} ,$$

$$p_1(h(\tau_{29})) = \frac{4412}{9801\sqrt{29}} \qquad \text{(by (67))} ,$$

$$p_2(h(\tau_{29})) = \frac{455\sqrt{29}}{60254729561664} \qquad \text{(by (73))} .$$

This gives

$$\frac{1}{\pi} = \alpha\left(\frac{2(i\sqrt{58}-58)}{\pi i \cdot 2} + \frac{(i\sqrt{58}-58)^2}{2} \cdot \frac{4 \cdot 455\sqrt{29}}{9801} \sum_{n=0}^{\infty} n\, c(n) \left(\frac{1}{396}\right)^{4n}\right)$$

$$+\beta\left(\frac{(-1)(i\sqrt{58}-58)^2}{2} \cdot \frac{4412}{9801\sqrt{29}} \sum_{n=0}^{\infty} c(n) \left(\frac{1}{396}\right)^{4n}\right). \tag{75}$$

Next we observe that for

$$29 \cdot 59 \cdot \left(\frac{1}{\pi} - \alpha\frac{i\sqrt{58}-58}{\pi i}\right)\left(\frac{(i\sqrt{58}-58)^2}{2}\right)^{-1} = \frac{1}{\pi}.$$

This turns (75) into

$$\frac{1}{\pi} = 29 \cdot \frac{4 \cdot 455\sqrt{2}}{9801} \sum_{n=0}^{\infty} n\, c(n) \left(\frac{1}{396}\right)^{4n} + \frac{4 \cdot 1103}{9801\sqrt{2}} \sum_{n=0}^{\infty} c(n) \left(\frac{1}{396}\right)^{4n},$$

which, by recalling $c(n)$ from (23), is Ramanujan's series (14).

This completes the description of the steps of the Sato construction. Fundamental ingredients of this construction are solutions to the Sato Eigenvalue Problem (Sato-EVP) forming Sato triples; recall Definition 2.3. Further aspects related to these notions are discussed in Sections 4, 5, and 6.

4. Infinite Families of Sato triples for Ramanujan-Gosper

The starting point, Step 0, of the Sato construction of Ramanujan's series (1) was made by the pair

(76)
$$g(\tau) := (1+\lambda(\tau))\theta_3(\tau)^4 \in M_2(\Gamma;\chi) \text{ and } h(\tau) := \frac{\lambda(\tau)(1-\lambda(\tau))^2}{16(1+\lambda(\tau))^4} \in M_0(\Gamma),$$

where

(77)
$$\Gamma := \left\langle \Gamma(2), \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & -2 \\ 1 & 0 \end{pmatrix} \right\rangle = \langle \Gamma(2), \gamma_1, \gamma_2 \rangle.$$

To execute Step 0 a local expansion $(g, h, Y; \Gamma)$ was determined algorithmically, as explained in Section 10; i.e., the computation of a series

(78)
$$Y(z) := \sum_{n=0}^{\infty} \underbrace{\frac{(1/4)_n(1/2)_n(3/4)_n}{(1)_n(1)_n n!} 256^n}_{=c(n)} \cdot z^n,$$

with holonomic coefficient sequence $(c(n))$, such that for τ with $\Im(\tau)$ sufficiently large,

$$g(\tau) = Y(h(\tau)).$$

To carry out all the remaining steps of the construction, Step 1 to Step 8, in principle, one only needs to choose a solution to the Sato-EVP; in our running example this was the Sato triple

(79)
$$(N, \gamma_N, \tau_N) = \left(29, \begin{pmatrix} 58 & -2 \\ 59 & -2 \end{pmatrix}, \frac{2}{59}\frac{58 + i\sqrt{58}}{58}\right)$$

for Γ as in (45). From this data, all the ingredients needed to establish Ramanujan's series (1) can be computed—primarily with the help of the algorithm MultiSamba.

A simple but far reaching observation is that a given group Γ can give rise to parameterized families of Sato triples with infinitely many members. The next lemma presents such a family which contains (79) as the special case with $N = 29$.

Lemma 4.1. *For each $N \in \mathbb{Z}_{\geq 2}$ one has that*

(80)
$$(N, \gamma_N, \tau_N) = \left(N, \begin{pmatrix} 2N & -2 \\ 2N+1 & -2 \end{pmatrix}, \frac{2}{2N+1}\frac{2N + i\sqrt{2N}}{2N}\right)$$

is a Sato triple for Γ as in (77). Moreover, for γ_1 and γ_2 as in (77),

(81)
$$\begin{pmatrix} 2N & -2 \\ 2N+1 & -2 \end{pmatrix} = \gamma_1\gamma_2\gamma_1^{-N}$$

and

(82)
$$\tau_N = \begin{pmatrix} 0 & -2 \\ 1 & -2N \end{pmatrix} i\sqrt{2N} \ \ with \ \begin{pmatrix} 0 & -2 \\ 1 & -2N \end{pmatrix} = \gamma_1^N \gamma_2.$$

Proof. The proof is by straight-forward verification. □

Note that the instance $N = 29$ of (81) and (82) corresponds to (11) and (42), respectively.

Example 4.2. The choice $N = 95$ in (80) gives the Sato triple,

(83)
$$(N, \gamma_N, \tau_N) = \left(95, \begin{pmatrix} 190 & -2 \\ 191 & -2 \end{pmatrix}, \frac{2}{191} \frac{190 + i\sqrt{190}}{190}\right).$$

Then by (82),
$$\tau_{95} = \begin{pmatrix} 0 & -2 \\ 1 & -190 \end{pmatrix} i\sqrt{190},$$
and for h as in (76) one has,

(84)
$$h(i\sqrt{190}) = \frac{1}{(12\sqrt{19}\,(481 + 340\sqrt{2}))^4} = \left(\frac{12\sqrt{19}\,(481 - 340\sqrt{2})}{7 \cdot 9 \cdot 16 \cdot 19 \cdot 23}\right)^4.$$

This h and the modular form g as in (76) together with the Sato triple (83) gives rise to the following approximating sum with $c(n)$ as in (78),

$$\phi(m) = \frac{\sqrt{19}}{2 \cdot 3^2 \cdot 7^2 \cdot 19 \cdot 23^2} \sum_{n=0}^{m} \Big(40(693121 + 5457\sqrt{2})n$$

(85)
$$+ (1877581 - 869892\sqrt{2})\Big) c(n) \left(\frac{12\sqrt{19}\,(481 - 340\sqrt{2})}{7 \cdot 9 \cdot 16 \cdot 19 \cdot 23}\right)^{4n}, \ m \geq 0.$$

If m tends to infinity this gives a series with limit $1/\pi$; i.e.,

(86)
$$\frac{1}{\pi} = \lim_{m\to\infty} \phi(m).$$

Remark 4.3. Remarkably the approximating sum $\phi(m)$ adds 16 correct digits when increasing m incrementally in steps of 1. For example,
$$\phi(0) = \frac{1877581 - 869892\sqrt{2}}{466578\sqrt{19}} = 0.318309886183790619\ldots,$$
gives 16,
$$\phi(1) = \frac{5\left(81308478120690184913 - 38307502735843794300\sqrt{2}\right)}{977797011746620033\,92\sqrt{19}}$$
$$= 0.31830988618379067153776752674502 7\ldots,$$

gives 32 correct digits of $1/\pi$, etc. The series (1) only adds 8 correct digits when adding the next summand. The celebrated Chudnovsky series [11, eq. (1.5)],

$$\frac{426880\sqrt{10005}}{\pi} = \sum_{n=0}^{\infty}(545140134n + 13591409)c(n)\left(\frac{-1}{640320}\right)^{3n} \tag{87}$$

adds less digits than $\phi(\infty)$, namely 14. This series has been used until today for world-record computations of digits of π; see [29].

Another aspect we want to stress is that the Sato construction with input (76), (77), (78), and the infinite family (80) of Sato triples, can be viewed as an algorithmic version of the following theorem by the Borweins:

Theorem 4.4 ([6, Thm. 5] and [4, eq. (5.5.16)])**.**

$$\frac{1}{\pi} = \sum_{n=0}^{\infty} \frac{(1/4)_n(1/2)_n(3/4)_n\, d_n(N)}{(n!)^3} x_N^{2n+1},$$

where,

$$x_N := \frac{4k_N(1-k_N^2)}{(1+k_N^2)^2} := \left(\frac{g_N^{12}+g_N^{-12}}{2}\right)^{-1},$$

with

$$d_n(N) = \left(\frac{\alpha(N)x_N^{-1}}{1+k_N^2} - \frac{\sqrt{N}}{4}g_N^{-12}\right) + n\sqrt{N}\left(\frac{g_N^{12}-g_N^{-12}}{2}\right),$$

$\alpha(N)$ as in(88)*, and*

$$k_N = \lambda^*(N) := \sqrt{\lambda(i\sqrt{N})}, \quad g_N^{12} = \frac{1-k_N^2}{2k_N}.$$

The entities g_N are called Ramanujan-Weber class invariants; they were studied independently by Ramanujan [25] and Weber [28]. The relation to one of the three Weber functions is as follows [4, eq. (3.2.13)],

$$g_N = 2^{-1/4} f_1(\sqrt{-N}).$$

The singular value function $\alpha(r)$, $r \in \mathbb{R}_{>0}$, was introduced by the Borweins in [4, eq. (5.1.1)]. The sections 5.1 to 5.3 of [4] are devoted to a study of its properties, including computational aspects. In [6, eq. (7.3)] a representation in terms of an x-series is given,

$$\alpha(r) = \left(\frac{1}{\pi} - 4\sqrt{r}\cdot\frac{\sum_{n=-\infty}^{\infty} n^2(-x_r)^{n^2}}{\sum_{n=-\infty}^{\infty}(-x_r)^{n^2}}\right)\left(\sum_{n=-\infty}^{\infty} x_r^{n^2}\right)^{-4}, \tag{88}$$

where $x_r := q_2(i\sqrt{r}) = e^{-\pi\sqrt{r}}$. As r tends to infinity, x_r tends to zero and $\alpha(r)$ tends to $1/\pi$. A crucial key in the work of the Borweins is to calculate $\alpha(N)$ for $N \in \mathbb{Z}_{>0}$ which is done iteratively.

Ramanujan's series (1) is obtained from Theorem 4.4 by setting $N = 58$.

It is interesting to have a glance at the corresponding comment made in [6], "For $N = 58$ [...] Ramanujan [25] and Weber [28] have calculated g_{58} for us [...] sophisticated number-theoretic techniques exist for computing k_N [...] Knowing α is equivalent to determining that the number 1103 [in Ramanujan's series (1)] is correct. It is less clear how one explicitly calculates $\alpha(58)$ in algebraic form, except by brute force, and a considerable amount of brute force is required; but a numerical calculation to any reasonable accuracy is easily obtained from (88) and 1103 appears!"

To complete the gap in proving their derivation of Ramanujan's series (1) (i.e., proving that 1103 is indeed correct), the Borweins set up a numerical estimation in way that Gosper's computation delivered sufficiently many digits of π to conclude that 1103 is indeed correct. The interested reader finds details of the Borwein reasoning in [6, sec. 8].

In contrast, the Sato construction as described above works purely on algebraic grounds. In other words, if the computational complexity of the problem allows that the required singular values $h_N := h(\tau_N)$, $p_1(h_N)$, and $p_2(h_N)$ can be computed by a computer algebra implementation of the MultiSamba algorithm, this at the same time is a guarantee for their correctness!

It is even possible to automate the Sato construction with regard to families of Sato triples. For example, for the Sato family (80), the values for $N \geq 2$ where $1/h(i\sqrt{2N})$ is an integer are:

$$N = 2, 3, 5, 9, 11, 29.$$

Hemmecke's implementation in FriCAS delivers for all the primes among these values the corresponding series for $1/\pi$; see

`https://www.risc.jku.at/people/hemmecke/papers/oneoverpi`

Remark 4.5. Hemmecke is working on an extension of his MultiSamba implementation from primes to arbitrary integers. This then will allow to treat also the case $N = 9$, and also the case $N = 95$ of the Sato family in (80) which then would deliver a computer proof of $\lim_{m\to\infty} \phi(m) = 1/\pi$ for the approximating sum $\phi(m)$ from (85).

5. Sato triples and Chudnovsky Families

In this section we present further infinite families of Sato triples where we take as the starting point, Step 0, of the Sato construction the pair

$$g(\tau) := \sqrt{E_4(\tau)} = 1 + 120q - 6120q^2 + 737760q^3 + O(q^4), \tag{89}$$

$$h(\tau) := \frac{1}{j(\tau)} = q - 744q^2 + 356652q^3 + O(q^4), \tag{90}$$

where $q = q_1(\tau) = e^{2\pi i \tau}$. Here $E_4(\tau)$ is the normalized Eisenstein series of weight 4, and $j(\tau)$ is the modular j-function (Felix Klein's j-invariant); i.e., we have

$$E_4(\tau) \in M_4(\Gamma) \text{ and } j(\tau) \in M_0(\Gamma) \text{ for } \Gamma := \mathrm{SL}_2(\mathbb{Z}). \tag{91}$$

As explained in [23, Sec. 5.1], the following local expansion $(g, h, Y; \Gamma)$ can be determined algorithmically:

$$Y(z) := \sum_{n=0}^{\infty} \underbrace{\frac{(1/6)_n(1/2)_n(5/6)_n}{(1)_n(1)_n n!} 1728^n}_{=c(n)} \cdot z^n = \sum_{n=0}^{\infty} \frac{(6n)!}{(3n)!(n!)^3} z^n \tag{92}$$

with holonomic coefficient sequence $(c(n))$, such that for τ with $\Im(\tau)$ sufficiently large,

$$g(\tau) = Y(h(\tau)).$$

Remark 5.1. Here, as well as for other applications, we need to consider a slightly more general setting for local expansions. Namely, when g is defined as a root of a modular form,

$$g(\tau) := f(\tau)^{1/r}$$

where $f \in M_{2r}(\Gamma)$ such that $f(\infty) \neq 0$. It turns out that also in this case a local expansion $(g, h, Y; \Gamma)$ exists with Y being again a power series with holonomic coefficients. As indicated in Section 10, such local expansions come from the existence of differential equations such as `Out[86]`. Yifan Yang's proof [30, Thm. 1] covers the special case $r = 1$. The proof for $g = f^{1/r}$ with $f \in M_{kr}(\Gamma)$, $k \in \mathbb{Z}_{\geq 1}$, is essentially the same.

Two infinite families of Sato triples for $\Gamma = \mathrm{SL}_2(\mathbb{Z})$ give rise to two infinite families of series for $1/\pi$.

5.1. **Chudnovsky family 1.** The following lemma in a special case gives rise to the celebrated series (87).

Lemma 5.2. *For each $N \in \mathbb{Z}_{\geq 2}$ one has that*

$$(N, \gamma_N, \tau_N) = \left(N, \begin{pmatrix} -1 & -1 \\ 1 & 0 \end{pmatrix}, \frac{-1 + i\sqrt{4N-1}}{2N}\right) \tag{93}$$

is a Sato triple for $\Gamma = \mathrm{SL}_2(\mathbb{Z})$. Moreover,

$$\tau_N = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \frac{1 + i\sqrt{4N-1}}{2}. \tag{94}$$

Proof. The proof is by straight-forward verification. □

Let us choose the local expansion $(g, h, Y; \Gamma)$ with (89), (90), and (92) as the starting point of the Sato construction. Then (93) gives rise to an infinite family of series for $1/\pi$. Note that by (94),

$$j(\tau_N) = j\left(\frac{1 + i\sqrt{4N-1}}{2}\right)$$

which evaluates to cubes of integers in case that $4N - 1$ is a Heegner number. The largest one is obtained with $N = 41$ for which

$$j\left(\frac{1 + i\sqrt{163}}{2}\right) = -640320^3,$$

and the corresponding series obtained by the Sato construction is (87).

To present one more member of this family, for $N = 17$ one has

$$j\left(\frac{1 + i\sqrt{67}}{2}\right) = -5280^3$$

and obtains

$$\frac{1760\sqrt{330}}{\pi} = \sum_{n=0}^{\infty}(10177 + 261702n)\frac{(1/6)_n(1/2)_n(5/6)_n}{(1)_n(1)_n n!}1728^n\left(\frac{-1}{5280}\right)^{3n} \tag{95}$$

as the result of the Sato construction. Each summand adds about the same number of correct digits to $1/\pi$ as the series (1).

5.2. **Chudnovsky family 2.** The following lemma, which is straight-forward to prove, presents another elementary family of Sato triples which induces another infinite "Chudnovsky family" of series for $1/\pi$.

Lemma 5.3. *For each $N \in \mathbb{Z}_{\geq 2}$ one has that*

$$(N, \gamma_N, \tau_N) = \left(N, \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \frac{i\sqrt{N}}{N}\right) \tag{96}$$

is a Sato triple for $\Gamma = \mathrm{SL}_2(\mathbb{Z})$.

This means, for τ_N as in (96),

$$\tau_N = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} i\sqrt{N}. \tag{97}$$

Again we choose the local expansion $(g, h, Y; \Gamma)$ with (89), (90), and (92) as the starting point of the Sato construction. Then (96) gives rise to another infinite family of series for $1/\pi$. Note that by (97),

$$j(\tau_N) = j(i\sqrt{N}),$$

which is an algebraic integer. At particular points one obtains integer values; for example [14, Table (12.20)],

$$j(i\sqrt{7}) = 255^3.$$

Carrying out the Sato construction for the Sato triple with $N = 7$, one obtains,

$$\frac{1}{\pi} = \frac{18\sqrt{255}}{7225} \sum_{n=0}^{\infty} (133n + 8)c(n)\Big(\frac{1}{255}\Big)^{3n}, \tag{98}$$

which was already recorded by Ramanujan [25].

As in the case of the Ramanujan-Gosper family (80), the Sato construction with input (89), (90) and (92), and the infinite families of Sato triples (93) and (96) can be viewed as an algorithmic version of a theorem:

Theorem 5.4 ([11, eq. (1.4)] and [12, eq. (4.9)])**.** *Let*

$$s_2(\tau) := \frac{E_4(\tau)}{E_6(\tau)} \left(E_2(\tau) - \frac{3}{\pi \Im(\tau)} \right)$$

where the $E_{2j}(\tau)$ *are the Eisenstein series as in* [19, eqs. (11), (12), (13)]. *Then for* $\tau = (1 + \sqrt{-d})/2$*:*

$$\sum_{n=0}^{\infty} \left(\frac{1 - s_2(\tau)}{6} + n \right) \cdot \frac{(6n)!}{(3n)!(n!)^3} \cdot \frac{1}{j(\tau)^n} = \frac{1}{\pi} \cdot \frac{(-j(\tau))^{1/2}}{(d(1728 - j(\tau))^{1/2}}. \tag{99}$$

The choice $d = 163$ gives (87), a member of the first Chudnovsky family induced by (93).

Despite the fact that the Chudnovskys explicitly restricted[1] the validity of (99) to points of the form $\tau = (1 + \sqrt{-d})/2$, with a mild modification their formula seems to hold also for the second Chudnovsky family induced by (96).

This modification is as follows: instead of restricting to $\tau = (1 + \sqrt{-d})/2$, one extends the domain of validity of (99) to all $\tau \in \mathbb{H}$ which are irrational quadratic numbers with negative discriminant $-d$. For example, the discriminant of $\tau_{41} =$

[1] E.g., in [12, p. 45] they state, "Expression (99) can be used whenever $\tau = (1 + \sqrt{-d})/2$."

$(-1+i\sqrt{163})/82$ is -163 since τ_{41} is a root of $41x^2+x+1$ which has discriminant -163. The discriminant of $(1+i\sqrt{163})/2$ being a root of x^2-x+41 is also -163. As an example for the suggested extension: the discriminant of $\tau := i\sqrt{7}$ is -28; substituting this τ into (99) with $d = 28$ gives (98), a series of the second Chudnovsky family!

Also for the Chudnovsky families 1 and 2 it is possible to automate the Sato construction; see

`https://www.risc.jku.at/people/hemmecke/papers/oneoverpi`

for corresponding $1/\pi$ series delivered by Hemmecke's FriCAS implementation.

6. Sato triples and Apéry-Beukers-Chan Families

The infinite family of Sato triples presented in this Section in a more direct way relates to the original work of Sato or, more precisely, to the work of Heng Huat Chan and collaborators [9] inspired by Sato.

As the starting point, Step 0, of the Sato construction we take the pair

$$g(\tau) := \prod_{n=1}^{\infty} \frac{(1-q^{2n})^7(1-q^{3n})^7}{(1-q^n)^5(1-q^{6n})^5} = \frac{\eta(2\tau)^7\eta(3\tau)^7}{\eta(\tau)^5\eta(6\tau)^5}, \tag{100}$$

$$h(\tau) := q\prod_{n=1}^{\infty} \frac{(1-q^n)^{12}(1-q^{6n})^{12}}{(1-q^{2n})^{12}(1-q^{3n})^{12}} = \frac{\eta(\tau)^{12}\eta(6\tau)^{12}}{\eta(2\tau)^{12}\eta(3\tau)^{12}}, \tag{101}$$

where $q = q_1(\tau) = e^{2\pi i\tau}$. We have

$$g(\tau) \in M_2(\Gamma) \text{ and } h(\tau) \in M_0(\Gamma) \text{ for } \Gamma := \Gamma_1(6). \tag{102}$$

As explained in [23, Sec. 5.2], this pair is taken from Beukers' modular-form-based proof [3] of the irrationality of $\zeta(3)$. The following local expansion $(g, h, Y; \Gamma)$ can be determined algorithmically [23, `Out[41]`]:

$$Y(z) := \sum_{n=0}^{\infty} \underbrace{\sum_{k=0}^{n} \binom{n}{k}^2 \binom{n+k}{k}^2}_{=c(n)} \cdot z^n \tag{103}$$

with holonomic coefficient sequence $(c(n))$, such that for τ with $\Im(\tau)$ sufficiently large,

$$g(\tau) = Y(h(\tau)).$$

Remark 6.1. This time the elements $c(n)$ of the holonomic coefficient sequence are the celebrated Apéry numbers which satisfy a linear recurrence of order two [23, `Out[41]`]. As stated by Beukers [3, p. 274], the function

$$y(\tau) := \frac{\eta(\tau)^4\eta(6\tau)^8}{\eta(2\tau)^8\eta(3\tau)^4} \in M_0(\Gamma)$$

is a Hauptmodul for $\Gamma = \Gamma_1(6)$; i.e., it generates the field of modular functions on $\Gamma_1(6)$. For example [3, Sec. 2, eq. (1)],

$$h(\tau) = y(\tau)\frac{1-9y(\tau)}{1-y(\tau)}, \tag{104}$$

which can be found and proven algorithmically as described in Subsection 12.3 using the procedure `GuessAE`.

6.1. **Apéry-Beukers-Chan family 1.** This time the N in the infinite family of Sato triples comes in parameterized form.

Lemma 6.2. *For $\ell \in \mathbb{Z}_{\geq 1}$ one has that*

$$(N, \gamma_N, \tau_N) = \left(6\ell+1, \begin{pmatrix} -6\ell+1 & -\ell \\ 6 & 1 \end{pmatrix}, \frac{-6\ell+i\sqrt{6\ell}}{6(6\ell+1)}\right) \tag{105}$$

is a Sato triple for $\Gamma = \Gamma_1(6)$. Moreover, for $N = 6\ell+1$,

$$\tau_N = \begin{pmatrix} 1 & 0 \\ -6 & 1 \end{pmatrix} i\sqrt{\ell/6}. \tag{106}$$

Proof. The proof is by straight-forward verification. □

Consequently, $N = 7$ implied by $\ell = 1$ will give the first series for $1/\pi$ as the result of the Sato construction based on the local expansion $(g, h, Y; \Gamma = \Gamma_1(6))$ with (100), (101), and (103) and the family (105) of Sato triples. Beukers [3, Prop. 2.1], using the relation (104), proved that

$$h\left(\frac{i}{\sqrt{6}}\right) = (\sqrt{2}-1)^4. \tag{107}$$

However, for this choice the series $Y\left((\sqrt{2}-1)^4\right)$ does not converge.

Hence the next candidate value from Lemma 6.2 is $N = 13$, implied by the choice $\ell = 2$. A closed form of the corresponding image $h_2 := h(i/\sqrt{3})$ is easily guessed algorithmically. We start with a numerical approximation;the first 10 digits will be sufficient:

In[37]:= $\mathbf{h_2 = 0.0192378864;}$

In[38]:= $\mathbf{RootApproximant[h_2]}$

Out[38]= $26 - 15\sqrt{3}$

Inspired by (101), we call a Mathematica function to try for an even simpler expression:

In[39]:= **ResourceFunction["RadicalDenest"][(26 − 15Sqrt[3])^1/3]**

Out[39]= 2 − Sqrt[3]

Summarizing, Mathematica suggests the value

$$h\left(\frac{i}{\sqrt{3}}\right) = (2-\sqrt{3})^3. \tag{108}$$

The corresponding series for $1/\pi$ is

$$\frac{1}{\pi} = \frac{6}{9+\sqrt{3}} \sum_{n=0}^{\infty} \left(2n+1-\frac{1}{\sqrt{3}}\right) c(n)(2-\sqrt{3})^{3n}, \tag{109}$$

where the $c(n)$ are the Apéry numbers from (103).

This series is the first instance $\ell = 2$ of an infinite family specified in a theorem by H.H. Chan and H. Verrill. It involves h as in (101), and an entity p_g whose non-trivial definition can be found in their paper.

Theorem 6.3 ([10, eq. (5.1)])**.** *Let* $t_0 := h(i/\sqrt{6\ell})$. *Then for* $\ell \in \mathbb{Z}_{\geq 2}$,

$$\frac{1}{\pi} = \frac{\sqrt{\ell}\sqrt{t_0^2 - 34t_0 + 1}}{\sqrt{6}} \sum_{n=0}^{\infty} (2n + p_g)\, c(n) t_0^n, \tag{110}$$

where the $c(n)$ *are the Apéry numbers from* (103).

Remark 6.4. Note that the invariance [3, p. 274], $h(-1/(6\tau)) = h(\tau)$, implies

$$h\left(i\sqrt{\frac{\ell}{6}}\right) = h\left(i\frac{1}{\sqrt{6\ell}}\right).$$

As noted by Chan and Verrill, the instance $\ell = 5$ (i.e., $N = 31$ in (105)) recovers Sato's original example [26, entry 1, p. 2],

$$\frac{1}{\pi} = \frac{4\sqrt{15}}{9+4\sqrt{5}} \sum_{n=0}^{\infty} \left(2n+1-\frac{3\sqrt{5}}{10}\right) c(n)(2-\sqrt{5})^{4n}, \tag{111}$$

where the $c(n)$ are the Apéry numbers from (103).

6.2. **Apéry-Beukers family 2.** We conclude this section with another infinite family of Sato triples different from (105). Despite the similar parameterization, it gives rise to a family of series for $1/\pi$ different from (110) but still involving the Apéry numbers in the coefficient sequences.

Lemma 6.5. *For $\ell \in \mathbb{Z}_{\geq 1}$ one has that*

$$(N, \gamma_N, \tau_N) = \left(6\ell + 1, \begin{pmatrix} -6\ell - 5 & -\ell - 1 \\ 6 & 1 \end{pmatrix}, \frac{-6\ell - 3 + i\sqrt{6\ell - 3}}{6(6\ell + 1)}\right) \tag{112}$$

is a Sato triple for $\Gamma = \Gamma_1(6)$. Moreover, for $N = 6\ell + 1$,

$$\tau_N = \begin{pmatrix} 1 & 0 \\ -6 & 1 \end{pmatrix} \left(\frac{1}{2} + \frac{i}{2}\sqrt{\frac{2\ell - 1}{3}}\right). \tag{113}$$

Proof. The proof is by straight-forward verification. □

Taking this family of Sato triples together with the local expansion $(g, h, Y; \Gamma = \Gamma_1(6))$ with (100), (101), and (103) as input for the Sato construction gives a family of series for $1/\pi$ different from (110). The first five relevant g values are:

$$g(\tau_7) = -1,\ g(\tau_{13}) = -(2 - \sqrt{3})^2,\ g(\tau_{19}) = -\left(\frac{3 - \sqrt{5}}{2}\right)^4,$$

$$g(\tau_{25}) = \left(\frac{\sqrt{21} - 5}{2}\right)^3,\ g(\tau_{31}) = -\left(-1 + 2^{1/3}\right)^4.$$

We restrict ourselves to state only one of the corresponding series for $1/\pi$; for the Sato triple (113) with $N = 25$, implied by $\ell = 4$, the Sato construction results in,

$$\frac{1}{\pi} = \frac{21\sqrt{3}}{14 + 9\sqrt{21}} \sum_{n=0}^{\infty} \left(4n + 2 - \frac{5\sqrt{21}}{21}\right) c(n) \left(\frac{\sqrt{21} - 5}{2}\right)^{3n}, \tag{114}$$

where the $c(n)$ are the Apéry numbers from (103).

Remark 6.6. At the time of the completion of the paper, Ralf Hemmecke started to work on an implementation of our method to automatically derive series for $1/\pi$ induced by Sato triples of the Apéry-Beukers families.

7. The algorithm MultiSamba illustrated

The algorithmic aspects of the Sato construction, as described above, involves the following algorithmic aspects:

(i) the holonomic toolbox, primarily for guessing but also for proving;
(ii) the algorithm ModFormDE for proving the correctness of a conjectured linear differential equation in derivatives of a modular form and with coefficients being polynomials in a modular function;
(iii) the main computational engine, the algorithm MultiSamba to derive and prove algebraic relations between modular functions.

Further information on aspects (i) and (ii) can be found in [22] and [23]. A description of the functionality of the algorithm MultiSamba is the object of this section. We restrict to presenting illustrating examples. A more detailed description of the underlying mechanism will be given in a separate publication.

7.1. **Working with expansions at the cusps.** When dealing with modular functions being analytic on $\mathbb{H}$, the MultiSamba algorithm works by representing them in the form of tuples of Fourier series expansions at the cusps.

Example 7.1. Recall the definition of the modular function $\lambda(\tau) \in M_0(\Gamma(2))$ from (20). For MultiSamba it is represented by a triple,

$$\lambda \leftrightarrow (\lambda_\infty, \lambda_0, \lambda_1),$$

where the entry λ_r is a truncated Fourier series representation of λ at the cusp r.

The representation by triples in the example is owing to the fact that the group $\Gamma = \Gamma(2)$ induces the three cusps $\infty, 0$, and 1. In general, if MultiSamba has to deal with modular functions in $M_0(\Gamma)$ where Γ induces n cusps, the functions are represented by n-tuples. The word "Multi" in MultiSamba stands for doing computations with such tuples, in this way extending the algorithm Samba ("subalgebra module basis algorithm") originating from [24] and [16], which works with standard Laurent series (in the Fourier variable).

The required order of truncation depends on the particular application. In the computer algebra system FriCAS, there is no need to precompute a truncation point, because it provides a lazy Laurent series implementation that computes coefficients at the time they are needed for the concrete computation.

Example 7.2. TASK. Prove algorithmically the classical relation (122) occuring in Section 8,

$$\frac{1}{j(\tau)} = \frac{(\lambda(\tau)-1)^2\lambda(\tau)^2}{256\,(\lambda(\tau)^2-\lambda(\tau)+1)^3}. \tag{115}$$

The functions $j(\tau)$ and $\lambda(\tau)$ are analytic on $\mathbb{H}$. Hence the proof of (115) reduces to the verification of the relation `rel=0` at the cusps, where

In[40]:= $\mathbf{rel = (L-1)^2L^2 * J - 256(L^2-L+1)^3;}$

Here J and L are symbols standing for the j and λ function, respectively. Next we input suitable representations of λ and j in terms of triples

$$\lambda \leftrightarrow (\lambda_\infty, \lambda_0, \lambda_1) \text{ and } j \leftrightarrow (j_\infty, j_0, j_1),$$

consisting of expansions at the cusps:

In[41]:= $\boldsymbol{\lambda_\infty = 16x - 128x^2 + 704x^3 - 3072x^4 + O[x]^5;}$

$\boldsymbol{\lambda_0 = 1 - 16x + 128x^2 - 704x^3 + 3072x^4 + O[x]^5;}$

$$\boldsymbol{\lambda_1 = -\frac{1}{16}x^{-1} + \frac{1}{2} - \frac{5}{4}x + \frac{31}{8}x^3 - \frac{27}{2}x^5 + O[x]^6;}$$
$$\boldsymbol{j_\infty = j_0 = j_1 = x^{-2} + 744 + 196884x^2 + 21493760x^4 + O[x]^6;}$$

the symbol x stands for $x = q_2 = e^{\pi i \tau}$.

Remark 7.3. In Hemmecke's FriCAS toolbox the expansions from `In[41]` are computed with the package QEta [17].

As operations on lists (recall: Mathematica uses "$\{\dots\}$" notation for lists), addition and multiplication are carried out componentwise; for example,

In[42]:= $\mathbf{L - 1\ /.L \to \{\lambda_\infty, \lambda_0, \lambda_1\}}$

Out[42]= $\{-1 + 16x - 128x^2 + 704x^3 - 3072x^4 + O[x]^5, -16x + 128x^2 - 704x^3 + 3072x^4 + O[x]^5, -1 - \frac{1}{16}x^{-1} + \frac{1}{2} - \frac{5}{4}x + \frac{31}{8}x^3 - \frac{27}{2}x^5 + O[x]^6\}$

In[43]:= $\mathbf{L^2\ /.L \to \{\lambda_\infty, \lambda_0, \lambda_1\}}$

Out[43]= $\{256x^2 - 4096x^3 + 38912x^4 - 278528x^5 + O[x]^6, 1 - 32x + 512x^2 - 5504x^3 + 45056x^4 + O[x]^6, \frac{1}{256}x^{-2} - \frac{1}{16}x^{-1} + \frac{13}{32} - \frac{5}{4}x + \frac{69}{64}x^2 + \frac{31}{8}x^3 - 8x^4 + O[x]^5\}$

This arithmetical feature of Mathematica makes the verification of `rel=0` particularly convenient:

In[44]:= $\mathbf{rel\ /.\{J \to \{j_\infty, j_0, j_1\}, L \to \{\lambda_\infty, \lambda_0, \lambda_1\}\}}$

Out[44]= $\{O[x]^4, O[x]^4, O[x]^1\}$

The meaning of this output list is this: in each of its three components the corresponding Laurent series involves no summand with non-negative power of x. In other words, this verifies that at each cusp the modular function

$$(\lambda(\tau) - 1)^2 \lambda(\tau)^2 \cdot j(\tau) - 256\left(\lambda(\tau)^2 - \lambda(\tau) + 1\right)^3,$$

which corresponds to the symbolic expression `rel`, has no pole and has no non-zero constant part in its x-expansion. This implies that it is zero, and (115) is proven.

Remark 7.4. If one would work with a description of j with smaller trunctation order, e.g.,

In[45]:= $\mathbf{j_\infty = j_0 = j_1 = x^{-2} + 744 + 196884x^2 + O[x]^4;}$

the output triple would contain a constant (unspecified) or x-powers of negative order. For example, in the concrete case of input `In[45]` one has,

In[46]:= $\mathbf{rel\ /.\{J \to \{j_\infty, j_0, j_1\}, L \to \{\lambda_\infty, \lambda_0, \lambda_1\}\}}$

Out[46]= $\{O[x]^4, O[x]^4, O[x]^0\}$

which would not suffice for a proof of `rel=0`. In other words, the user has to take care of choosing truncation orders of sufficient size.

7.2. **Computing algebraic relations.** Given modular functions $\phi, \psi \in M_0(\Gamma)$ which are analytic on $\mathbb{H}$, in this section we informally describe how the algorithm MultiSamba computes a polynomial $p(X, Y)$, with coefficients in a suitable extension field $\mathbb{K}$ of $\mathbb{Q}$, such that $p(\phi(\tau), \psi(\tau)) = 0$.

As explained in the previous Section 7.1, instead of working directly with the functions ϕ and ψ, MultiSamba operates on representatives which are n-tuples of formal Laurent series in x, n being the number of cusps induced by Γ. Recall that analytically $x = q_w = e^{2\pi i\tau/w}$, but owing to a problem transformation onto algebraic grounds, MultiSamba interprets x as a symbolic variable.

To compute $p(X, Y)$, MultiSamba proceeds by using a *reduction* mechanism: Let $f = (f_1, \ldots, f_n)$ be an n-tuple of formal Laurent series in x. Let $B = \{\mathbf{1}, b_0, b_1, ..., b_k\}$ be a finite set of n-tuples, each b_j being a formal Laurent series in x and $\mathbf{1}$ standing for the n-tuple $(1, \ldots, 1)$. A *reduction* of f with respect to B is an n-tuple $(g_1, \ldots, g_n)$ defined as

$$(g_1, \ldots, g_n) := f - (\alpha\mathbf{1} + \beta_0\, b_0 + \cdots + \beta_k\, b_k) \tag{116}$$

with $\alpha, \beta_j \in \mathbb{K}$.

The notion of *reduction* is justified by using a special order relation,

$$(f_1, \ldots, f_n) \succeq (g_1, \ldots, g_n). \tag{117}$$

Its precise specification is of quite technical nature; it will be presented in a separate paper.

Remark 7.5. A more elementary reduction relation would be to define (117) whenever the rightmost non-zero entry in

$$(\operatorname{ord} f_1 - \operatorname{ord} g_1, \operatorname{ord} f_2 - \operatorname{ord} g_2, \ldots, \operatorname{ord} f_n - \operatorname{ord} g_n)$$

is negative ("reverse lexicographic ordering"). This "$\succeq$" reduction would indeed work for the next Example 7.6, which can be easily checked in each particular step. But, as already indicated, the reduction to make MultiSamba work in general is more complicated and requires also division by the order of b_0, kind of a special element in the reduction process; in the Example 7.6, $b_0 = J = (j_\infty, j_0, j_1)$ and since $(\operatorname{ord} j_\infty, \operatorname{ord} j_0, \operatorname{ord} j_1) = (-2, -2, -2)$ (equal entries), division can be avoided. The actual '$\succeq$" used my MultiSamba looks at the component that contains the smallest order (breaking ties by selecting bigger components first). Hence "reverse lexicographic order" in general is not applicable.

Example 7.6. We use the MultiSamba algorithm to derive the algebraic relation,

$$(\lambda(\tau) - 1)^2\lambda(\tau)^2 \cdot j(\tau) - 256\left(\lambda(\tau)^2 - \lambda(\tau) + 1\right)^3 = 0, \tag{118}$$

which is equivalent to (115).

Corresponding to `In[41]` of Example 7.2, we first input triples L and J as representatives of λ and j, respectively.

In[47]:= $\mathbf{L = \{16x - 128x^2 + 704x^3 - 3072x^4 + O[x]^5, 1 - 16x + 128x^2 - 704x^3 + 3072x^4 + O[x]^5, -\frac{1}{16}x^{-1} + \frac{1}{2} - \frac{5}{4}x + \frac{31}{8}x^3 - \frac{27}{2}x^5 + O[x]^6\};}$

In[48]:= $\mathbf{j_\infty = j_0 = j_1 = x^{-2} + 744 + 196884x^2 + 21493760x^4 + O[x]^6;}$

In[49]:= $\mathbf{J = \{j_\infty, j_0, j_1\};}$

In[50]:= $\mathbf{b_0 = J;}$

As components of L we chose again the λ_r from `In[41]`.

In the last line we defined $b_0 := J$. Trying to reduce L with respect to $\{1, b_0\}$ fails, which is obvious in view of $\operatorname{ord} j_1 = -2$ in contrast to $\operatorname{ord} l_1 = -1$.

As a repeated step of the algorithm, we take the result of the reduction, which in this case is L itself, as an additional reduction element b_1,

In[51]:= $\mathbf{b_1 = L;}$

Then we try to reduce $L * b_1$ w.r.t. $\{1, b_0, b_1\}$:

In[52]:= $\mathbf{L * b_1}$

Out[52]= $\{256x^2 - 4096x^3 + O[x]^4, 1 - 32x + O[x]^2, \frac{1}{256}x^{-2} - \frac{1}{16}x^{-1} + O[x]^0\}$

MultiSamba finds the reduction,

In[53]:= $\mathbf{L * b_1 - \frac{1}{256}b_0 - b_1}$

Out[53]= $\{-\frac{1}{256}x^{-2} - 93/32 + O[x]^1, -\frac{1}{256}x^{-2} - 93/32 + O[x]^1, -3 - 768x^2 + O[x]^3\}$

Again we take the result of the reduction as an additional reduction element b_2,

In[54]:= $\mathbf{b_2 = L * b_1 - \frac{1}{256}b_0 - b_1;}$

Next we try to reduce $L * b_2$ w.r.t. $\{1, b_0, b_1, b_2\}$:

In[55]:= $\mathbf{L * b_2}$

Out[55]= $\{-\frac{1}{16}x^{-1} + \frac{1}{2} + O[x]^1, -\frac{1}{256}x^{-2} + \frac{1}{16}x^{-1} + O[x]^0, \frac{3}{16}x^{-1} - 3/2 + O[x]^1\}$

MultiSamba finds the reduction,

In[56]:= $\mathbf{L * b_2 - \frac{1}{256}b_0 - b_1}$

Out[56]= $\{-\frac{1}{16}x^{-1} + \frac{1}{2} + O[x]^1, -\frac{1}{256}x^{-2} + \frac{1}{16}x^{-1} + O[x]^0, 48x - 384x^2 + O[x]^3\}$

Again we take the result of the reduction as an additional reduction element b_3,

In[57]:= $\mathbf{b_3 = L * b_2 - \frac{1}{256}b_0 - b_1;}$

Then we try to reduce $L * b_3$ w.r.t. $\{1, b_0, b_1, b_2, b_3\}$. This reduction fails, and we take the result of the reduction, which is $L * b_3$ itself, as an additional reduction element b_4,

In[58]:= $\mathbf{b_4 = L * b_3;}$

Next we try to reduce $L * b_4$ w.r.t. $\{1, b_0, b_1, b_2, b_3, b_4\}$:

In[59]:= $\mathbf{L * b_4}$

Out[59]= $\{-16x + 384x^2 + O[x]^3, -\frac{1}{256}x^{-2} + \frac{3}{16}x^{-1} + O[x]^0, \frac{3}{16}x^{-1} - \frac{9}{2} + O[x]^1\}$

MultiSamba finds the reduction,

In[60]:= $\mathbf{L * b_4 - b_4 - (-3)b_1}$

Out[60]= $\{1 + 16x + O[x]^2, \frac{1}{16}x^{-1} + \frac{1}{2} + O[x]^1, 16x + O[x]^2\}$

Again we take the result of the reduction as an additional reduction element b_5,

In[61]:= $\mathbf{b_5 = L * b_4 - b_4 - (-3)b_1;}$

Next we try to reduce $L * b_5$ w.r.t. $\{1, b_0, b_1, b_2, b_3, b_4, b_5\}$:

In[62]:= $\mathbf{L * b_5}$

Out[62]= $\{16x + 128x^2 + O[x]^3, \frac{1}{16}x^{-1} - \frac{1}{2} + O[x]^1, -1 + O[x]^1\}$

MultiSamba finds the reduction,

In[63]:= $\mathbf{L * b_5 - b_5 + 1}$

Out[63]= $\{O[x]^4, O[x]^3, O[x]^1\}$

This means,

$$0 = L\, b_5 - b_5 + \mathbf{1}.$$

With backwards substitution one obtains,

$$\begin{aligned}
0 &= L\, b_5 - b_5 + \mathbf{1} = (L - \mathbf{1})b_5 + \mathbf{1} \\
&= (L - \mathbf{1})\left((L - \mathbf{1})b_4 - (-3)b_1\right) + \mathbf{1} \\
&= (L - \mathbf{1})\left((L - \mathbf{1})L\, b_3 + 3b_1\right) + \mathbf{1} \\
&= (L - \mathbf{1})\left((L - \mathbf{1})L\,(L\, b_2 + 3b_1) + 3b_1\right) + \mathbf{1} \\
&= (L - \mathbf{1})^2 L^2 b_2 + 3(L - \mathbf{1})(L^2 - L + \mathbf{1})b_1 + \mathbf{1} \\
&= (L - \mathbf{1})^2 L^2 \left(L\, b_1 - b_1 - \frac{1}{256} b_0\right) + 3(L - \mathbf{1})(L^2 - L + \mathbf{1})b_1 + \mathbf{1} \\
&= -\frac{1}{256}(L - \mathbf{1})^2 L^2 b_0 + \left((L - \mathbf{1})^3 L^2 b_1 + 3(L - \mathbf{1})(L^2 - L + \mathbf{1})b_1 + \mathbf{1}\right) \\
&= -\frac{1}{256}(L - \mathbf{1})^2 L^2 \cdot J + \left((L - \mathbf{1})^3 L^3 + 3(L - \mathbf{1})(L^2 - L + \mathbf{1})L + \mathbf{1}\right).
\end{aligned}$$

Observing that

$$(L - \mathbf{1})^3 L^3 + 3(L - \mathbf{1})(L^2 - L + \mathbf{1})L + \mathbf{1} = (L^2 - L + \mathbf{1})^3,$$

implies

$$p(X, Y) = -\frac{1}{256}(X - 1)^2 X^2 \cdot Y + (X^2 - X + 1)^3, \tag{119}$$

which completes the derivation of (118).

We conclude this section with a couple of remarks. First, once an algebraic relation is derived with MultiSamba, this derivation stands also for a correctness proof of this relation.

Second, in the reduction process of Example 7.6 the variable $b_0 = J$ arises in only one reduction relation, `In[53]`; hence Y occurs only linearly in (119). This is owing to the fact that λ is a Hauptmodul for $\Gamma(2)$. In general, the variable b_0 will arise in more than one reduction relation and thus Y will occur non-linearly in the output polynomial $p(X, Y)$.

Finally, the proof that one can specify an order relation (117) such that MultiSamba reduction works and terminates also in general will be presented in a separate article.

8. Proof of $h(\tau_{29}) = 1/396^4$ and related computational aspects

An immediate proof of (54) is obtained by substituting the value

$$\lambda(i\sqrt{58}) = (13\sqrt{58} - 99)^2(\sqrt{2} - 1)^{12}, \tag{120}$$

entry (33) of [20], into the definition (50) of λ.

At the end of Section 3.7 we indicated a hybrid numeric-symbolic strategy for closed form evaluation of $h(\tau_N)$. To obtain an algebraic form of $\lambda(i\sqrt{58})$, one could, for example, proceed as follows. Again we use the Mathematica built-in function `ModularLambda` for modular λ. In the first step we compute the desired value with a 100-digit precision.

In[64]:= **ML = ModularLambda;**

In[65]:= **val1 = N[ML[I Sqrt[58]], 100]**

Out[65]= 6.506377248432371405123936314707760832880425765816861600622453228853543540741472816728773788040682328 $\times 10^{-10}$

Next we use a Mathematica procedure which suggests a polynomial having a root located "as close as possible" to the numerical approximation `val1` of $\lambda(i\sqrt{58})$:

In[66]:= **p = MinimalPolynomial[RootApproximant[val1], z]**

Out[66]= $1 - 1536953612\,z + 3073907238\,z^2 - 1536953612\,z^3 + z^4$

The built-in solver computes four roots represented in terms of nested radicals. Out of those we pick that one which is equal to the expression on the right of (120):

In[67]:= **val2 = Solve[p == 0, z, Quartics → True][[1, 1, 2]]**

Out[67]= $384238403 - 50452974\sqrt{58} - 13860\sqrt{2\left(768555217 - 100916244\sqrt{58}\right)}$

In[68]:= **val2 − ((13$\sqrt{58}$ − 99)^2($\sqrt{2}$ − 1)^{12}) //FullSimplify**

Out[68]= 0

We remark that there is a Mathematica function which tries to "denest" nested radicals:

In[69]:= **ResourceFunction["RadicalDenest"][val2]**

Out[69]= $384238403 - 271697580\sqrt{2} + 71351280\sqrt{29} - 50452974\sqrt{58}$

A correctness proof of the root representation `val2` can be obtained by using the classical evaluation for the modular j function,

$$j(i\sqrt{58}) = 30^3(140989 + 26163\sqrt{29})^3, \tag{121}$$

entry (58) of [21], together with the classical relation

$$j(\tau) = 2^8 \frac{(1-\lambda(\tau)+\lambda(\tau)^2)^3}{\lambda(\tau)^2(1-\lambda(\tau))^2}, \tag{122}$$

entry (2) of [21].

Remark 8.1. The evaluation (121) is classical in the following sense, as described at [21]. There are 18 numbers d having class number $h(-d) = 2$, where $h(-d)$ is the class number of the binary quadratic form discriminant $-d$ of the quadratic field $\mathbb{Q}(\sqrt{-d})$. The evaluation (121) falls into the group of six entries [21, (53) to (59)] where d is of the form $d = 4m$; the case $m = 58$ corresponds to (121).

The computational correctness proof goes as follows:

In[70]:= $\mathbf{val3 = 30^3(140989 + 26163\sqrt{29})^3;}$

In[71]:= $\left(\frac{\mathbf{2^8\,(1-L+L^2)^3}}{\mathbf{(L-1)^2L^2}}\mathbf{/.L \to val2}\right) - \mathbf{val3\ //FullSimplify}$

Out[71]= 0

Algorithmic discovery and proof of (122) using the MultiSamba algorithm is discussed in Section 7.

Algorithmic discovery of (121) can be done analogously to the finding of an algebraic representation of $\lambda(i\sqrt{58})$ as shown before.

8.1. MultiSamba discovers and proves $h(i\sqrt{58}) = 1/396^4$.

We conclude this section by describing how to derive,

$$h(\tau_{29}) = h(i\sqrt{58}) = \left(\frac{1}{396^4}\right),$$

with the help of Hemmecke's implementation of the MultiSamba algorithm.

For our running Example 2.6 we consider the function $\iota(\tau) := \frac{1}{h(\tau)}$, see (17). It is easy to verify that $\iota(\tau)$ does not have any pole in $\mathbb{H}$; expansions at the cusps can be easily obtained. The same holds true for the function $\iota(29\tau)$. Analogously to our description in Section 9, with the help of MultiSamba we can find a polynomial $f \in \mathbb{Z}[x,y]$ such that $f(\iota(\tau), \iota(29\tau) = 0$ for every $\tau \in \mathbb{H}$. In particular for τ_{29} as given by (10), we have $\iota(\tau_{29}) = \iota(29\tau_{29})$. Therefore, $\iota(\tau_{29})$ must be a root of the polynomial $f(x,x)$. This polynomial is of degree 58 and has the following primitive factors over $\mathbb{Q}$:

$$
\begin{aligned}
&x - 24591257856, \quad x - 2509056, \quad x - 2304, \quad x - 648, \quad x - 81, \quad x,\\
&x + 1024, \quad x + 3969, \quad x + 12288, \quad x + 82944, \quad x + 6635520,\\
&x^2 - 695296512x + 691798081536, \quad x^2 - 16584912x + 69257922561,\\
&x^2 + 68825088x + 21743271936, \quad x^2 + 3553320960x - 770527199232,\\
&x^2 + 19990130688x + 52205595918336,\\
&x^3 - 10618854144x^2 - 204506726400x - 19677812097024,\\
&x^3 + 22252544x^2 - 11204034560x + 2578054119424,\\
&x^4 - 1265184x^3 + 1624487296x^2 - 89027692544x + 7815289901056.
\end{aligned}
\tag{123}
$$

Note that according to (42), we have $\iota(\tau_{29}) = \iota(i\sqrt{58})$. One can find rational numbers l_x and u_x such that $l_x < x_\infty := e^{-\pi\sqrt{58}} < u_x$; this is done by using upper and lower bounds from $\mathbb{Q}$ sufficiently close to $\pi\sqrt{58}$, and then truncating the exponential series expansion appropriately.

With the help of these bounds we can find rational upper and lower bounds of $\lambda(i\sqrt{58})$ by using the first few terms of the defining θ-functions according to (20). Then, using (17), we can find rational numbers l and u such that $l < \iota(i\sqrt{58}) < u$.

By the above construction, it is clear that we can make the interval (l,u) as small as desired. In particular, we can make it so small that there is only one factor from (123) that has a root in (l,u). Furthermore, this interval should be made so small that there is only one of the roots of this factor in (l,u). Checking the number of real roots of a polynomial inside a given interval can be done constructively by applying Sturm's Theorem [27, Ex. 4.32].

Finally, since $l < \iota(i\sqrt{58}) < u$ and there is exactly one real root of one particular factor from (123) in the interval (l,u), this root must be equal to $\iota(i\sqrt{58})$. In our case, the interval (l,u) is such that the first factor from (123) is selected, which eventually leads to $h(\tau_{29}) = \frac{1}{396^4}$.

All the above steps are automated in the QEta package, i.e. the value $h(\tau_N)$ can be found algorithmically from a Sato triple.

9. Algorithmic computation of $p_1(h(\tau_N))$ and $p_2(h(\tau_N))$

Using the algorithm MultiSamba one cannot only find but also prove algebraic representations of $p_1(h(\tau_N))$ and $p_2(h(\tau_N))$. In this section we present the MultiSamba method by describing how to derive and prove the closed form representations in (67) of Lemma 3.18 and in (73) of Lemma 3.22. In the following two sections N is specified as $N := 29$, but to make the general idea more transparent we keep the generic variable N in.

9.1. **Computation of** $p_1(h(\tau_N))$**.** The general idea is to derive a low degree polynomial $p \in \mathbb{Z}[x]$ such that $p_1(h(\tau_N))$ is a root of p.

Recall the definition of $g \in M_2(\Gamma;\chi)$ and $h \in M_0(\Gamma)$ from (76) for the group Γ given by (77). Furthermore, by Lemma 3.16 we know $\frac{H(\tau)}{g(\tau)} \in M_0(\Gamma'_\chi)$. We also have $h(\tau), h(N\tau) \in M_0(\Gamma'_\chi)$. Consequently, there are polynomials $f_1(x,y)$, $f_2(x,y) \in \mathbb{Z}[x,y]$ such that

$$f_1\left(h(\tau), \frac{H(\tau)}{g(\tau)}\right) = 0 \text{ and } f_2\left(h(N\tau), \frac{H(\tau)}{g(\tau)}\right) = 0 \text{ for any } \tau \in \mathbb{H}.$$

By the Sato triple property we have $h(\tau_N) = h(N\tau_N)$. Hence it follows that $\frac{H(\tau_N)}{g(\tau_N)}$ must be a root of the polynomial $p(x) := \gcd(f_1(h(\tau_N),x), f_2(h(\tau_N),x))$.

Our goal is to compute such modular polynomials f_1 and f_2 using the algorithm MultiSamba which is implemented in FriCAS as part of Hemmecke's QEta package. In general, such computations are rather technical and time consuming.

Moreover, to bring MultiSamba into position, some preprocessing is required. As explained in Section 7, MultiSamba represents a modular function by its expansions at the cusps; in addition, the modular functions in MultiSamba have to be analytic on $\mathbb{H}$. Consequently, poles in $\mathbb{H}$ must be removed first. Therefore, we used $\iota(\tau) := \frac{1}{h(\tau)}$ and $K(\tau) := \frac{29}{24}\iota(\tau)\iota(N\tau)\frac{H(\tau)}{g(\tau)}$ instead of $\frac{H(\tau)}{g(\tau)}$, and find f_1 and f_2 by computing the modular polynomials $\bar{f}_1, \bar{f}_2$ such that

$$\bar{f}_1(\iota(\tau), K(\tau)) = \bar{f}_2(\iota(N\tau), K(\tau)) = 0,$$

respectively. using MultiSamba. In fact, we compute

$$\bar{p}(x) = \gcd(\bar{f}_1(\iota(\tau_N),x), \bar{f}_2(\iota(\tau_N),x)).$$

For the particular case of our running Example 2.6 we have $\iota(\tau_N) = 396^4$ and get $\deg(\bar{f}_1(396^4,x)) = \deg(\bar{f}_2(396^4,x)) = 60$ and $\deg(\bar{p}(x)) = 2$. While the monic polynomials $\bar{f}_1(396^4,x), \bar{f}_2(396^4,x)$ have coefficients with up to about 1400 digits, their greatest common divisor is

$$\bar{p}(x) = x^2 - 37310298054509454034264668636445629480 96.$$

By taking the positive root and undoing the multiplication by $\iota(\tau_N)\iota(N\tau_N) = 396^8$, we obtain $p_1(h(\tau_{29})) = \frac{4412\sqrt{29}}{284229}$.

For further computational details ; see
`https://www.risc.jku.at/people/hemmecke/papers/oneoverpi`

9.2. **Algorithmic computation of** $p_2(h(\tau_N))$**.** Recall Lemma 3.20 where we proved

$$\frac{h'(\tau)}{g(\tau)} \in M_0(\Gamma_\chi) \text{ with } \Gamma_\chi := \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma : \chi(a,b,c,d) = 1 \right\} \leq \Gamma,$$

and Γ as in (77). Also, $h(\tau) \in M_0(\Gamma_\chi)$, hence in view of $h(\tau_N) = 1/396^4$ we aim at computing a polynomial $f \in \mathbb{Z}[x,y]$ such that $f(h(\tau), p_2(h(\tau)) = 0$.

As in the previous section, to bring MultiSamba into position, we first remove all poles, again by using $\iota(\tau) := \frac{1}{h(\tau)}$. Then we invoke MultiSamba to compute instead of f the related modular polynomial $\bar{f}$ such that

$$\bar{f}(\iota(\tau), L(\tau)) = 0 \text{ for } L(\tau) := \iota(\tau)^2 p_2(h(\tau)).$$

For our running Example 2.6 it turns out that this polynomial is of degree 2 and equal to $\bar{f}(x,y) = y^2 - x(x-256)$. By substituting $x = 396^4$, taking the positive root and undoing the multiplication by $\iota(\tau_{29})^2 = 396^8$, we get $p_2(h(\tau_{29})) = \frac{455\sqrt{29}}{60254729561664}$.

For further computational details ; see
`https://www.risc.jku.at/people/hemmecke/papers/oneoverpi`

10. Algorithmic derivation of local expansion (24)

The algorithmic discovery of local expansions using the holonomic toolbox is described in detail in [23]. To obtain (24), one proceeds as follows.

```
In[72]:= Theta3[N_] := Series[EllipticTheta[3, 0, x], {x, 0, N}];
In[73]:= Theta3[5]^4
Out[73]= 1 + 8x + 24x^2 + 32x^3 + 24x^4 + 48x^5 + O[x]^6
```

Instead of invoking Mathematica's built-in functions such as `EllipticTheta`, in various cases one may prefer alternative definitions:[2]

```
In[74]:= Theta2to4[N_] := Module[{s},
           s[M_] := Sum[x^(n(n+1)), {n, 0, M}]^4; Series[16x s[N], {x, 0, N}]]
In[75]:= Theta2to4[9]
```

[2]E.g., as here, to exploit sparsity of terms.

Out[75]= $16x + 64x^3 + 96x^5 + 128x^7 + 208x^9 + O[x]^{10}$

In[76]:= $\textbf{lam}[\textbf{N}\_] := \dfrac{\textbf{Theta2to4}[\textbf{N}]}{\textbf{Theta3}[\textbf{N}]^4}$

In[77]:= **lam[5]**

Out[77]= $16x - 128x^2 + 704x^3 - 3072x^4 + 11488x^5 + O[x]^6$

Next we input the x-series for g and h; see also (21) and (22):

In[78]:= $\textbf{h}[\textbf{N}\_] := \dfrac{\textbf{lam}[\textbf{N}](1 - \textbf{lam}[\textbf{N}])^2}{16(1 + \textbf{lam}[\textbf{N}])^4}$

In[79]:= **h[5]**

Out[79]= $x - 104x^2 + 6444x^3 - 311744x^4 + 13018830x^5 + O[x]^6$

In[80]:= $\textbf{g}[\textbf{N}\_] := (1 + \textbf{lam}[\textbf{N}])\textbf{Theta3}[\textbf{N}]^4$

In[81]:= **g[5]**

Out[81]= $1 + 24x + 24x^2 + 96x^3 + 24x^4 + 144x^5 + O[x]^6$

Now we are ready to compute the first twelve coefficients $c(0)$ to $c(11)$ such that (24):

In[82]:= **comp[N_] := ComposeSeries[g[N], InverseSeries[h[N]]];**
cli = CoefficientList[Normal[comp[11]], x]

Out[82]= {1, 24, 2520, 369600, 63063000, 11732745024, 2308743493056, 472518347558400,
99561092450391000, 21452752266265320000, 4705360871073570227520,
1047071828879079131681280}

To carry out the next step, we load Mallinger's package `GeneratingFunctions` written in Mathematica, which is freely available at `https://combinatorics.risc.jku.at/software`. This RISC package implements a holonomic-tool box for guessing and proving; see [23] for further details:

In[83]:= **<< RISC'GeneratingFunctions'**

Version 0.8 written by Christian Mallinger © RISC-JKU

Based on the first twelve values $c(0)$ to $c(11)$ as input, Mallinger's procedure `GuessRE` produces a *guess* of a recurrence the holonomic coefficient sequence $(c(n))_{n\geq 0}$ might satisfy such that (24) holds:

In[84]:= **cRE = GuessRE[cli, c[n]][[1]]**

Out[84]= $\{-8(1 + 2n)(1 + 4n)(3 + 4n)c[n] + (1 + n)^3 c[1 + n] = 0, c[0] = 1\}$

Applying `RSolve`, a recurrence solver coming with the Mathematica system, produces a closed form for the $c(n)$:

In[85]:= **RSolve[cRE, c[n], n][[1]]**

Out[85]= $\{c[n] \to (3 \times 2^{8n-5} \dfrac{\text{Pochhammer}[\frac{5}{4}, -1+n]\text{Pochhammer}[\frac{3}{2}, -1+n]\text{Pochhammer}[\frac{7}{4}, -1+n]}{\text{Pochhammer}[2, -1+n]^3}\}$

which matches (23).

So far, the recurrence for the $c(n)$ and their closed form (23) were derived as *conjectures*. But, as explained in [22] and [23], there is an algorithmic method, the algorithm ModFormDE, which delivers a computer-assisted proof for this. This proof is based on holonomic transformation and on zero recognition for modular functions. More concretely, the transformation consists in rewriting the recurrence `cRE` for the $c(n)$ into an equivalent form, namely, into a differential equation for the generating function

$$Y(z) = \sum_{n=0}^{\infty} c(n) z^n.$$

This step is carried out automatically as follows:

In[86]:= **cDE = RE2DE[cRE, c[n], Y[z]]**

Out[86]= $\{-24\mathtt{Y}[\mathtt{z}] - (-1 + 816\mathtt{z})\mathtt{Y}'[\mathtt{z}] - 3(-\mathtt{z} + 384\mathtt{z}^2)\mathtt{Y}''[\mathtt{z}] - (-\mathtt{z}^2 + 256\mathtt{z}^3)\mathtt{Y}^{(3)}[\mathtt{z}] = 0,$
$\mathtt{Y}[0] = 1, \mathtt{Y}'[0] = 24, \mathtt{Y}''[0] = 5040\}$

Solving this differential equation with Mathematica's `DSolve`, confirms the closed form (23):

In[87]:= **DSolve[cDE, Y[z], z][[1]]**

Out[87]= `{Y[z] → HypergeometricPFQ[{1/4, 1/2, 3/4}, {1, 1}, 256z]}`

Summarizing, by (4) we know there exists a local expansion such that for all $\tau \in \mathbb{H}$ with $\Im(\tau)$ sufficiently large,

$$g(\tau) = Y(h(\tau)) \text{ with } Y(z) = \sum_{n=0}^{\infty} c(n) z^n$$

and $(c(n))_{n \geq 0}$ a holonomic sequence. Using the holonomic toolbox we arrived at the conjecture that the $c(n)$ satisfy the recurrence `Out[84]`. This conjecture in turn is equivalent to the statement that the generating function $Y(z)$ satisfies the differential equation `Out[86]`. As described in [23] and [23] one can prove the validity of such conjectured differential equation algorithmically. The underlying idea in short is this: one takes the differential equation part of `Out[86]` in the analytic version

$$\begin{aligned} &-24Y(h(\tau)) - (-1 + 816h(\tau))Y'(h(\tau)) - 3(-h(\tau) + 384h(\tau)^2)Y''(h(\tau) \\ &\quad - (-h(\tau)^2 + 256h(\tau)^3)Y^{(3)}(h(\tau)) \end{aligned}$$

and, using

$$Y(h(\tau)) = g(\tau), Y'(h(\tau)) = \frac{g'(\tau)}{h'(\tau)}, \text{ a.s.o.,}$$

rewrites it into a form involving g and h and their derivatives only. Then one rewrites the resulting expression into "Yang form" [22, Lemma 4.7]; i.e., as a linear combination of functions with uniquely determined coefficients being modular functions for Γ. Finally, showing this linear combination to be zero amounts to algorithmic zero recognition of the modular function coefficients. Details are given in [22].

11. Neighborhoods for $g(\tau) = Y(h(\tau))$

Task (59) of Step 6 of the Sato construction is to determine a bound $L > 0$ such that

$$g(\tau) = Y(h(\tau)) \text{ with } Y(z) = \sum_{n=0}^{\infty} c(n) z^n \text{ holds for all } \Im(\tau) > L.$$

In this section we use our running example to illustrate how this can be done.

For convenience we display the required representations of the functions involved. For $x = e^{\pi i \tau}$,

$$\theta_2(\tau) = \sum_{n \in \mathbb{Z}+1/2} x^{n^2} = 2x^{1/4} \sum_{n=0}^{\infty} x^{n^2+n}, \tag{124}$$

and

$$\theta_3(\tau) = \sum_{n \in \mathbb{Z}} x^{n^2} = 1 + 2\sum_{n=1}^{\infty} x^{n^2}, \tag{125}$$

Recall,

$$g(\tau) := (1 + \lambda(\tau))\theta_3(\tau)^4 \text{ and } h(\tau) := \frac{\lambda(\tau)(1-\lambda(\tau))^2}{16(1+\lambda(\tau))^4},$$

where

$$\lambda(\tau) = \frac{\theta_2(\tau)^4}{\theta_3(\tau)^4}. \tag{126}$$

In (64) we stated a local expansion with neighborhood of validity,

$$g(\tau) = \sum_{n=0}^{\infty} \frac{(1/4)_n (1/2)_n (3/4)_n}{(1)_n (1)_n n!} 256^n h(\tau)^n, \ \Im(\tau) > 1.87. \tag{127}$$

In this section we want to prove that one indeed can choose the bound

$$L := 1.87. \tag{128}$$

As a first observation, one can quickly see that

$$L \geq 1. \tag{129}$$

Proof of (129). By inspection, $h(\tau)$ has a pole at $\tau_h := 1 + i$ since

$$\lambda(1+i) = -1;$$

just set $\tau = i$ in the classic transformations

$$\lambda(\tau + 1) = \frac{\lambda(\tau)}{\lambda(\tau) - 1} \text{ and } \lambda\left(-\frac{1}{\tau}\right) = 1 - \lambda(\tau),$$

which are FunGrim [18] properties bbfb6c and 07bf27, respectively.

Owing to the fact that $\lambda(\tau)$ is analytic on $\mathbb{H}$, all other poles of $h(\tau)$ are images of τ_h under the action of matrices

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \Gamma = \left\langle \Gamma(2), \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & -2 \\ 1 & 0 \end{pmatrix} \right\rangle.$$

All these images have imaginary part ≤ 1 because of

$$\Im\Big(\begin{pmatrix} a & b \\ c & d \end{pmatrix} \tau_h \Big) = \frac{ad-bc}{|c\tau_h + d|^2} \cdot \Im(\tau_h) = \frac{ad-bc}{(c+d)^2+c^2} \cdot \Im(\tau_h) \leq \Im(\tau_h) = 1.$$

This proves (129). □

Another observation concerns a criterion for $|256h(\tau)| < 1$ to make the series in (127) converge:

Lemma 11.1. *Let*

$$u := 5 + 4\sqrt{2} - 2\sqrt{2(7+5\sqrt{2})} = 0.0470219\dots \tag{130}$$

and

$$l := 5 + 4\sqrt{2} + 2\sqrt{2(7+5\sqrt{2})} = 21.26668\dots .$$

Then for all $\tau \in \mathbb{H}$ such that

$$|\lambda(\tau)| < u \ \textit{ or } \ |\lambda(\tau)| > l, \tag{131}$$

one has

$$|256\ h(\tau)| < 1. \tag{132}$$

Proof of Lemma 11.1. Let $\Im(\tau) > 1$ and $M := |\lambda(\tau)|$, then

$$|256\ h(\tau)| = \Big| 16 \frac{\lambda(\tau)(1-\lambda(\tau))^2}{(1+\lambda(\tau))^4} \Big| \leq 16 \frac{M(1+M)^2}{(1-M)^4}.$$

We want,

$$16\frac{M(1+M)^2}{(1-M)^4} < 1.$$

One has,

$$16M(1+M)^2 - (1-M)^4 = p_1(-p_2),$$

where

$$p_1 = M^2 - 2(5+4\sqrt{2})M + 1 \text{ and } p_2 = M^2 - 2(5-4\sqrt{2})M + 1.$$

Now, in view of the simple facts,

$$-p_2 \leq -1 \text{ for } M > 0, \ \text{ and } \ p_1 = (M-u)(M-l),$$

one has for $M > 0$,

$$16\frac{M(1+M)^2}{(1-M)^4} < 1 \ \text{ iff } \ p_1(-p_2) < 0 \ \text{ iff } \ p_1 > 0 \ \text{ iff } \ (M < u \text{ or } M > l).$$

This completes the proof of Lemma 11.1. □

Finally, in view of Lemma 11.1, we obtain a desired bound $L \geq 1$ by choosing L such that

$$|\lambda(\tau)| < u \text{ for all } \tau \in \mathbb{H} \text{ with } \Im(\tau) > L. \tag{133}$$

Indeed, it turns out that such L exists.

Proposition 11.2. *For $L = 1.87$ the upper bound u as in* (133) *holds.*

Proof. By (126) the inequality for u in (133) is equivalent to

$$|\theta_2(\tau)| < u^{1/4}|\theta_3(\tau)|, \tag{134}$$

where $u^{1/4}$ denotes the real fourth root; i.e.,

$$u^{1/4} = 0.46566654962266\ldots .$$

By (124),

$$|\theta_2(\tau)| = \left|2x^{1/4}\sum_{n=0}^{\infty} x^{n^2+n}\right| < 2|x|^{1/4}\sum_{n=0}^{\infty}|x|^{n^2+n} < 2|x|^{1/4}\sum_{k=0}^{\infty}|x|^k = \frac{2|x|^{1/4}}{1-|x|}.$$

By (125),

$$|\theta_3(\tau)| = \left|1+2\sum_{n=1}^{\infty} x^{n^2}\right| > |1+2x| - 2\left|\sum_{n=2}^{\infty} x^{n^2}\right| > |1+2x| - 2\sum_{n=2}^{\infty}|x|^{n^2}$$
$$> |1+2x| - 2|x|^4\sum_{k=0}^{\infty}|x|^k = |1+2x| - \frac{2|x|^4}{1-|x|} > 1-2|x| - \frac{2|x|^4}{1-|x|}$$

As a consequence, to guarantee (134) and thus (133), it suffices to choose $L \geq 1$ such that for all $x = e^{\pi i\tau}$ where $\tau = a + ib$ with $b > L$,

$$\frac{2|x|^{1/4}}{1-|x|} < u^{1/4}\left(1-2|x| - \frac{2|x|^4}{1-|x|}\right). \tag{135}$$

In view of

$$X := |x| = |e^{\pi i(a+ib)}| = e^{-\pi b} < e^{-\pi L} \leq e^{-\pi} = 0.04321\ldots ,$$

we can restrict to assume

$$0 < X = |x| < 27/625 = 0.0432 < e^{-\pi}. \tag{136}$$

To determine the real solutions $X := |x|$ to the inequality (135) we run Mathematica's cylindrical algebraic decomposition (CAD) procedure:

In[88]:= $\mathbf{u = 5 + 4\sqrt{2} - 2\sqrt{2\left(7+5\sqrt{2}\right)};}$

In[89]:= **CylindricalDecomposition** $\left[\frac{\mathbf{2X^{1/4}}}{\mathbf{1-X}} < \mathbf{u^{1/4}}\left(\mathbf{1-2X-\frac{2X^4}{1-X}}\right)\mathbf{, X}\right]$

Out[89]= $(0 \le X < R) \lor (X > 1)$

In[90]:= **N[R, 20]**

Out[90]= 0.0028401640299899030414

Because of the constraint (136), the only remaining solution interval is

$$0 < X < R.$$

Mathematica allows to determine the upper bound R with arbitrary precision; in `Out[90]` we computed 20 digits.

Summarizing, all $\tau = a + ib$ with real a and $b > 0$ such that $e^{-\pi b} < R$ satisfy the inequality (133). Observing that

$$b = -\frac{1}{\pi} \ln R = 1.8665352634873658736 9\ldots$$

confirms that one can choose

$$L = 1.87$$

to guarantee (133). This completes the proof of Proposition 11.2, and thus also (127) is proven. □

Remark 11.3. Mathematica represents the upper bound R with the data

$$\text{Root}[\{P_1, P_2\}, \{3, 1\}] \tag{137}$$

where

$$P_1 = P_1(z_1) = 1 - 4z_1^2 - 2z_1^4 - 4z_1^6 + z_1^8$$

and

$$P_2 = P_2(z_1, z_2) = z_1^4 - 16z_2 - 12z_1^4 z_2 + 62z_1^4 z_2^2 - \cdots + 16z_1^4 z_2^{16}.$$

The data (137) specifies R as follows. By inspection one sees that $P_1(z_1)$ has four real roots which are ordered according to size as $r_1 < r_2 < r_3 < r_4$. The 3 in the pair $\{3, 1\}$ of (137) tells us to take r_3 and to solve $P_2(r_3, z_2) = 0$ for z_2. This equation has two real solutions s_1 and s_2 such that $s_1 < s_2$. Finally, the 1 in the pair $\{3, 1\}$ of (137) tells us to take s_1 to obtain R; i.e.,

$$R = s_1 = \text{ smallest real root of } P_2(r_3, z_2).$$

12. Proof of Lemma 3.14: Modularity of g and h

Recall

$$\Gamma := \langle \Gamma(2), \gamma_1, \gamma_2 \rangle \text{ with } \gamma_1 := \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \text{ and } \gamma_2 := \begin{pmatrix} 0 & -2 \\ 1 & 0 \end{pmatrix}.$$

12.1. **Proof of $g \in M_2(\Gamma;\chi)$.** Let

$$g(\tau) := (1+\lambda(\tau))\theta_3(\tau)^4.$$

We will show that

$$g(\gamma\tau) = \chi(\gamma)\det(\gamma)^{-1}(c\tau+d)^2 g(\tau) \text{ for all } \gamma \in \langle \Gamma(2), \gamma_1, \gamma_2 \rangle, \tag{138}$$

where

$$\chi(\gamma_1) = 1, \chi(\gamma_2) = -1, \text{ and } \chi(\gamma) = 1 \text{ for all } \gamma \in \Gamma(2),$$

This means, to prove (40),

$$g \in M_2(\Gamma;\chi),$$

it is sufficient to prove (138) for the additional generators γ_1 and γ_2. To this end, we will use properties listed in the FunGrim library at `www.fungrim.org`.

FunGrim property 099301 for λ gives,

$$\lambda(\gamma_1\tau) = \frac{1}{\lambda(\tau)}. \tag{139}$$

FunGrim property 4d8b0f for $\theta_j(z,\tau)$ gives,

$$\theta_3(\gamma_1\tau)^4 = (\tau+1)^2\theta_2(\tau)^4. \tag{140}$$

Consequently,

$$\begin{aligned} g(\gamma_1\tau) &= (1+\lambda(\gamma_1\tau))\theta_3(\gamma_1\tau)^4 = \left(1+\frac{1}{\lambda(\tau)}\right)(\tau+1)^2\theta_2(\tau)^4 \\ &= (\tau+1)^2\left(1+\frac{\theta_3(\tau)^4}{\theta_2(\tau)^4}\right)\theta_2(\tau)^4 = (\tau+1)^2(1+\lambda(\tau))\theta_3(\tau)^4 \\ &= (\tau+1)^2 g(\tau). \end{aligned} \tag{141}$$

With regard to γ_2 the proof is as follows,

$$\lambda(\gamma_2\tau) = \lambda\left(\frac{-2}{\tau}\right) = \lambda\left(-\frac{1}{\tau/2}\right) = \lambda\left(\begin{pmatrix}0 & -1\\ 1 & 0\end{pmatrix}\frac{\tau}{2}\right) = 1-\lambda\left(\frac{\tau}{2}\right), \tag{142}$$

where the last equality is by FunGrim property 07bf27.

FunGrim property c4b16c for $\theta_3(z,\tau)$ gives,

$$\theta_3(\gamma_2\tau)^4 = -\left(\frac{\tau}{2}\right)^2\theta_3\left(\frac{\tau}{2}\right)^4. \tag{143}$$

Consequently,

$$g(\gamma_2\tau) = (1+\lambda(\gamma_2\tau))\theta_3(\gamma_2\tau)^4 = -\left(\frac{\tau}{2}\right)^2\left(2-\lambda\left(\frac{\tau}{2}\right)\right)\theta_3\left(\frac{\tau}{2}\right)^4.$$

To complete the proof we will show that

$$g(\gamma_2\tau) = -\frac{1}{\det(\gamma_2)}\tau^2 g(\tau), \tag{144}$$

which is equivalent to showing

$$\left(2-\lambda\left(\frac{\tau}{2}\right)\right)\theta_3\left(\frac{\tau}{2}\right)^4 = 2(1+\lambda(\tau))\theta_3(\tau)^4.$$

Using the definition of λ in terms of theta functions, we rewrite this as

$$2\theta_3\left(\frac{\tau}{2}\right)^4 - \theta_2\left(\frac{\tau}{2}\right)^4 = 2\theta_3(\tau)^4 + 2\theta_2(\tau)^4.$$

Applying FunGrim property 69b32e for $\theta_3(z,\tau/2)$ this equality turns into

$$2\left(\theta_3(\tau)^2+\theta_2(\tau)^2\right)^2 - \theta_2\left(\frac{\tau}{2}\right)^4 = 2\theta_3(\tau)^4 + 2\theta_2(\tau)^4,$$

which reduces to

$$4\theta_2(\tau)^2\theta_3(\tau)^2 = \theta_2\left(\frac{\tau}{2}\right)^4.$$

This equality can be derived by using FunGrim properties for $\theta_j(0,\tau)$ again:

$$\begin{aligned} 4\theta_2(\tau)^2\theta_3(\tau)^2 &= 4^2\frac{\eta(\tau)^6}{\theta_4(\tau)^2} \quad \text{(by FunGrim 557b19)} \\ &= 4^2\frac{\eta(\tau)^8}{\eta(\tau/2)^4} \quad \text{(by FunGrim 9448f2)} \\ &= \theta_2\left(\frac{\tau}{2}\right)^4 \quad \text{(by a9c825).} \end{aligned}$$

This completes the proof of (144), and by combining this with (141), also (40) and (138) are proven.

12.2. **Proof of $h \in M_0(\Gamma)$.** Let

$$h(\tau) := \frac{\lambda(\tau)(1-\lambda(\tau))^2}{16(1+\lambda(\tau))^4}.$$

We will show that

$$h(\gamma\tau) = h(\tau) \text{ for all } \gamma \in \langle\Gamma(2),\gamma_1,\gamma_2\rangle. \tag{145}$$

The invariance property (145) for $\gamma \in \Gamma(2)$ is implied by the classical $\Gamma(2)$-invariance of λ. Consequently, to prove the remaining part of (52),

$$h \in M_0(\Gamma),$$

it is sufficient to prove the modular transformation property (145) for the extra generators γ_1 and γ_2.

Applying (139) to the definition of h gives,

$$h(\gamma_1\tau) = \frac{\lambda(\tau)^{-1}(1-\lambda(\tau)^{-1})^2}{16(1+\lambda(\tau)^{-1})^4} = h(\tau).$$

Applying (142) gives,

$$h(\gamma_2\tau) = \frac{\left(1-\lambda\left(\frac{\tau}{2}\right)\right)\lambda\left(\frac{\tau}{2}\right)^2}{16\left(2-\lambda\left(\frac{\tau}{2}\right)\right)^4};$$

i.e., to show $h(\gamma_2\tau) = h(\tau)$ is equivalent to proving that for $z = \lambda(\tau)$ and $y = \lambda(2\tau)$,

$$\begin{aligned} 0 &= \frac{(1-z)z^2}{(2-z)^4} - \frac{y(1-y)^2}{(1+y)^4} \\ &= \frac{(1-z-y)(1+zy-y)\left(z^2y^2-2z^2y+z^2+16zy-16y\right)}{(2-z)^4(y+1)^4}. \end{aligned}$$

The verification of the respective modular equation of level 2:

(146) $\lambda(\tau)^2\lambda(2\tau)^2 - 2\lambda(\tau)^2\lambda(2\tau) + \lambda(\tau)^2 + 16\lambda(\tau)\lambda(2\tau) - 16\lambda(2\tau) = 0,\ \tau \in \mathbb{H},$

can be done (algorithmically) by classical methods which completes the proof of $h \in M_0(\Gamma)$.

Remark 12.1. The modular equation (146) can be found and proven algorithmically; see the next subsection 12.3.

12.3. **Finding and proving** (146) **algorithmically.** The algorithmic discovery of (146) can be done as follows. Taking as input the first 15 terms of $y := \lambda(2\tau) = \sum_{n=1}^{\infty} a(n)x^n$ and $z := \lambda(\tau) = \sum_{n=1}^{\infty} b(n)x^n$, $x = \exp(\pi i\tau)$, will be sufficient:

```
In[91]:= y15 = 16x^2 − 128x^4 + 704x^6 − 3072x^8 + 11488x^10 − 38400x^12
            + 117632x^14 + O[x]^15;
In[92]:= z15 = 16x − 128x^2 + 704x^3 − 3072x^4 + 11488x^5 − 38400x^6 + 117632x^7 − 335872x^8
            + 904784x^9 − 2320128x^10 + 5702208x^11 − 13504512x^12 + 30952544x^13
            − 68901888x^14 + O[x]^15;
In[93]:= comp[N_] := ComposeSeries[y15, InverseSeries[z15]];
         cli = CoefficientList[Normal[comp[15]], x]
```

Out[93]= $\left\{0, 0, \frac{1}{16}, \frac{1}{16}, \frac{7}{128}, \frac{3}{64}, \frac{165}{4096}, \frac{143}{4096}, \frac{1001}{32768}, \frac{221}{8192}, \frac{12597}{524288}, \frac{11305}{524288}, \frac{81719}{4194304}, \frac{37145}{2097152}, \frac{4345965}{268435456}\right\}$

```
In[94]:= GuessAE[cli, y[z]][[1]]
```

Out[94]= $\{z^2 + (-16 + 16z - 2z^2)y[z] + z^2y[z]^2 = 0, y[0] = 0\}$

This output means that

$$\begin{aligned} 0 &= \lambda(\tau)^2 + \left(-16 + 16\lambda(\tau) - 2\lambda(\tau)^2\right)\lambda(2\tau) + \lambda(\tau)^2\lambda(2\tau)^2 \\ &= \lambda(\tau)^2\lambda(2\tau)^2 - 2\lambda(\tau)^2\lambda(2\tau) + \lambda(\tau)^2 + 16\lambda(\tau)\lambda(2\tau) - 16\lambda(2\tau), \end{aligned}$$

where the last line is nothing but (146).

The proof of this algorithmic discovery of (146) can be done using the algorithm ModFormDE as described in [22] and [23]. The basis for this kind of proof is the differential equation obtained by Alternatively, one can verify (146) by checking the local expansions at the cups; an example of this method can be found in Section 7.1.

13. Conclusion

In the Introduction we mentioned Zudilin's article [31] entitled "Ramanujan-type formulae for $1/\pi$: A second wind?" and expressed our hope that the methods presented in this article will bring new "algorithmic wind" into this area. We want to stress that our hope is not restricted to the topic discussed in this paper, but also extends to a variety of other possible areas where the algorithm MultiSamba, or its interplay with holonomic methods and the algorithm ModFormDE, may provide new possibilites for research.

Acknowledgement. Special thanks go to Professor Heng Huat Chan for sharing a copy of Sato's abstract [26] and related historical data. Silviu Radu was funded in whole or in part by the Austrian Science Fund (FWF) 10.55776/PAT1332123.

References

[1] Richard Askey. Pi and the AGM. By Jonathan M. Borwein and Peter B. Borwein (book review). *The American Math. Monthly*, 95(9):895–890, 2009.

[2] Nayandeep Deka Baruah, Bruce C. Berndt, and Heng Huat Chan. Ramanujan's series for $1/\pi$: a survey. *The American Math. Monthly*, 116(7):567–587, 2009.

[3] Frits Beukers. Irrationality proofs using modular forms. *Astérisque*, 147–148:271–283, 1987.

[4] Jonathan M. Borwein and Peter B. Borwein. *Pi and the AGM.* John Wiley & Sons, 1987.

[5] Jonathan M. Borwein and Peter B. Borwein. Ramanujan and pi. *Scientific Amer.*, 256:112–117, 1988.

[6] Jonathan M. Borwein, Peter B. Borwein, and David H. Bailey. Ramanujan, modular equations, and approximations to pi or how to compute one billion digits of pi. *Amer. Math. Monthly*, 96(3):201–219, 1989.

[7] Heng H. Chan and Shaun Cooper. Rational analogues of Ramanujan's series for $1/\pi$. *Math. Proc. Camb. Phil. Soc*, 153:2:361–383, 2012.

[8] Heng Huat Chan. Personal communication. Email, October 17, 2023.

[9] Heng Huat Chan, Song Heng Chan, and Zhiguo Liu. Domb's numbers and Ramanujan-Sato type series for $1/\pi$. *Adv. Math*, 186:396–410, 2004.

[10] Heng Huat Chan and Helena Verrill. The Apéry numbers, the Almkvist-Zudilin numbers and new series for $1/\pi$. *Math. Res. Lett.*, 16(3):405–420, 2009.

[11] David V. Chudnovsky and Gregory V. Chudnovsky. Approximation and complex multiplication according to Ramanujan. In G. E. Andrews, R. A. Askey, B. C. Berndt, K. G. Ramanathan, and R. A. Rankin, editors, *Ramanujan Revisited*, pages 375–472, Boston, 1988. Academic Press.

[12] David V. Chudnovsky and Gregory V. Chudnovsky. Use of computer algebra for diophantine and differential equations. In R.D. Jenks, editor, *Computer Algebra, LNS in Pure and Appl. Math., vol. 113*, pages 1–81, New York, 1989. Dekker.

[13] Shaun Cooper. *Ramanujan's Theta Functions*. Springer, 2017.

[14] David A. Cox. *Primes of the Form $x^2 + ny^2$*. John Wiley & Sons, 1989.

[15] FriCAS team. FriCAS—an advanced computer algebra system, 2024. Available at `https://fricas.github.io/`.

[16] Ralf Hemmecke. Dancing samba with Ramanujan partition congruences. *Journal of Symbolic Computation*, 84:14–24, 2018.

[17] Ralf Hemmecke. QEta—A FriCAS package to compute with q-expansions of modular functions, 2025. Available at `https://risc.jku.at/sw/qeta` and `https://hemmecke.github.io/qeta`.

[18] Fredrik Johansson. Fungrim: a symbolic library for special functions. In A.M. Bigatti, J. Carette, J. Davenport, M. Joswig, and T. de Wolff, editors, *Mathematical Software — ICMS 2020, LNS in Computer Science Vol. 12097*, pages 315–323, Cham, 2020. Springer. Available at: `https://arxiv.org/pdf/2003.06181.pdf`.

[19] Wolfram MathWorld. Eisenstein Series. `https://mathworld.wolfram.com/EisensteinSeries.html`.

[20] Wolfram MathWorld. Elliptic Lambda Function. `https://mathworld.wolfram.com/EllipticLambdaFunction.html`.

[21] Wolfram MathWorld. j-Function. `https://mathworld.wolfram.com/j-Function.html`.

[22] Peter Paule and Cristian-Silviu Radu. An algorithm to prove holonomic differential equations for modular forms. In Alin Bostan and Kilian Raschel, editors, *Transcendence in Algebra, Combinatorics, Geometry and Number Theory*, pages 367–420, Cham, 2021. Springer.

[23] Peter Paule and Cristian-Silviu Radu. Holonomic relations for modular functions and forms: First guess, then prove. *Internatl. J. Number Theory*, 17:713–759, 2021.

[24] Cristian-Silviu Radu. An algorithmic approach to Ramanujan-Kolberg identities. *Journal of Symbolic Computation*, 68, Part 1:225–253, 2015.

[25] Srinivasa Ramanujan. Modular equations and approximations to π. *Quart. J. Math.*, 45:350–372, 1914.

[26] Takeshi Sato. Apéry numbers and Ramanujan's series for $1/\pi$ (abstract, 2 pages in Japanese). Talk presented at the Annual meeting of the Mathematical Society of Japan, March 28–31, 2002.

[27] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, 1999.

[28] Heinrich Martin Weber. *Lehrbuch der Algebra, Vol. 3 [1998]*. AMS Chelsea Publishing (3rd ed.), 1981.

[29] Wikipedia. Chudnovsky algorithm. `https://en.wikipedia.org/wiki/Chudnovsky_algorithm`.

[30] Yifan Yang. On differential equations satisfied by modular forms. *Math. Z*, 246:1–19, 2004.

[31] Wadim Zudilin. Ramanujan-type formulae for $1/\pi$: A second wind? In N. Yui, H. Verrill, and C.F. Doran, editors, *Modular Forms and String Duality. Fields Inst. Comm. 54*, pages 179–188, Providence, R.I., 2008. Amer. Math. Soc.

Research Institute for Symbolic Computation (RISC), Johannes Kepler University, A-4040 Linz, Austria

Email address: Ralf.Hemmecke@risc.jku.at

Research Institute for Symbolic Computation (RISC), Johannes Kepler University, A-4040 Linz, Austria, and Center for Applied Mathematics, Tianjin University, Tianjin 300072, China

Email address: Peter.Paule@risc.jku.at

Research Institute for Symbolic Computation (RISC), Johannes Kepler University, A-4040 Linz, Austria

Email address: Silviu.Radu@risc.jku.at