Advanced Studies in Pure Mathematics 21, 1992
Zeta Functions in Geometry
pp. 359–408

On Eigenvalues of the Laplacian for Hecke Triangle Groups

Dennis A. Hejhal*

Abstract.

The purpose of this Note is to provide a survey of some of the recent experimental work aimed at computing eigenvalues of the Laplacian for a variety of Hecke triangle groups $\mathbf{G}(2\cos\frac{\pi}{N}) \setminus H$.[1]

§1. Introduction

Let Γ be a Fuchsian group of finite area acting on the Poincaré upper half-plane H. Let

$$\Delta u = y^2(u_{xx} + u_{yy})$$

be the non-Euclidean Laplacian on H. One of the most important zeta functions associated with $\Gamma \setminus H$ is the Selberg zeta function $Z_\Gamma(s)$, which we'll simply write as $Z(s)$. It is well-known ([9, p.72 (11)], [10, p.498], [30, pp.75–79]) that the nontrivial zeros of $Z(s)$ are intimately connected with the spectral decomposition of Δ over $L_2(\Gamma \setminus H)$.

One would very much like to find a good way of computing $Z(s)$ for arbitrary Γ when s is restricted, say, to $\{-1 \leqq \mathrm{Re}(s) \leqq 2\}$.

Since Z has order $\leqq 2$, one possible approach would be to use the Hadamard product formula (cf., for instance, [9, pp.72 (10), 148 (10.1)], [10, pp.435–440, 496–499], [29] and [38]) to reduce things to calculating the *zeros* of $Z(s)$.[2]

Received July 1, 1990. Revised August 16, 1991.
*With support from NSF Grant DMS 89-10744 and computer time from the Minnesota Supercomputer Institute (CRAY2), Pittsburgh Supercomputing Center (YMP-832), and San Diego Supercomputer Center (XMP-48).

[1] Part of the work described herein is discussed (or, otherwise pursued) to greater depth in the author's forthcoming *American Mathematical Society Memoir* [12].

[2] Even if another method based on (say) numerical analytic continuation ultimately proved to be superior, apriori knowledge of the zeros of $Z(s)$ would still be quite useful, particularly as a *check* on the overall accuracy.

One knows, of course, that any zeros of $Z(s)$ situated along $\{\mathrm{Re}(s) = \frac{1}{2}\}$ must correspond to eigenvalues of $-\Delta$ under the mapping

$$\frac{1}{2} \pm iR \longrightarrow \frac{1}{4} + R^2,$$

at least for $R > 0$.

For this reason, as well as one of intrinsic importance, it would be highly desirable to find good ways of computing the discrete eigenvalues of $-\Delta$ (for given Γ).

From the intrinsic standpoint, it would be even better if one could somehow determine the corresponding eigen*functions* themselves.

Currently, not too much is known about either problem. Cf. [2], [10, Appendix C], [11], [16], [35], [39] for a quick look at some of the existing work. Papers [11, 39] are of particular interest here.

In cases where Γ *has cusps*, the philosophy of Sarnak and Phillips ([3, 23, 24, 28]) suggests that, unless Γ possesses some type of arithmeticity or symmetry property, the discrete spectrum of $-\Delta$ will be very sparse (and, most likely, finite).

⟦In order to recover the familiar Weyl law in such cases, one needs to (properly) combine the poles of the Eisenstein series in $\{\mathrm{Re}(s) < \frac{1}{2}\}$ with the discrete eigenvalues $\frac{1}{4} + R_n^2$. Cf. [10, pp.210, 231 (top), 437, 456–458, 476] and [32].⟧

In the present paper, we shall be concerned exclusively with the classical eigenvalues $\frac{1}{4}+R_n^2$ — and shall take as our Γ a Hecke triangle group $\mathbf{G}(2\cos\frac{\pi}{N})$. This choice represents one of the simplest possible generalizations of (the classical modular group) $\mathrm{PSL}(2,\mathbf{Z})$. Cf. [8, pp.592, 629].

For the sake of clarity, we stop to recall a few basic properties of $\mathbf{G}(2\cos\frac{\pi}{N})$. First and foremost: $\mathbf{G}_N \equiv \mathbf{G}(2\cos\frac{\pi}{N})$ is generated by

$$E(z) = -\frac{1}{z} \qquad \text{and} \qquad T(z) = z + \mathcal{L}$$

where $\mathcal{L} = 2\cos(\frac{\pi}{N})$. The number N is a positive integer $\geqq 3$. It is easily seen that

$$\mathcal{F}_N = H \cap \{|z| > 1\} \cap \left\{|\mathrm{Re}(z)| < \frac{\mathcal{L}}{2}\right\}$$

is a fundamental region for $\mathbf{G}_N \setminus H$ and that $\mathbf{G}_N$ has signature

$$(g, n; \nu_1, \dots, \nu_n) = (0, 3; 2, N, \infty).$$

Cf. [10, pp.5, 569], [19, pp.227, 235], [8, pp.609–616]. The group $\mathbf{G}_N$ is (thus) a particular realization of the Schwarz triangle group $\mathbf{T}(\frac{\pi}{2}, \frac{\pi}{N}, \frac{\pi}{\infty})$. It virtually goes without saying here that $\mathbf{G}_3 = \mathrm{PSL}(2, \mathbf{Z})$ and that $\mathbf{G}_N$ admits an obvious symmetry with respect to the imaginary axis.

The group $\mathbf{G}_N$ is known to be *commensurable* with $\mathrm{PSL}(2, \mathbf{Z})$ iff $N = 3, 4, 6$. Cf. [17] and [20, 36] for the "if" and "only if", respectively.

Our *goal* is to study the discrete spectrum of $\mathbf{G}_N \setminus H$ for a variety of N.

The results we describe (in §§5–6) will serve to amplify the earlier work of A. Winkler [39].

Winkler's approach is substantially different than ours (resting, as it does, on a fair number of preliminary lemmas).

Prior to outlining the "mechanics" of *our* approach, it is worthwhile to highlight what the Sarnak-Phillips philosophy specifically predicts concerning $\Gamma = \mathbf{G}(2\cos\frac{\pi}{N})$.

The group $\mathbf{G}_N$ has a symmetry with respect to the imaginary axis. Exactly as in [10, p.590 (13)], one finds that the spectral decomposition of $L_2(\mathbf{G}_N \setminus H)$ splits into two "halves", one "even" and one "odd." The Eisenstein series appears *only* in the "even" half.

The *odd* portion of $L_2(\mathbf{G}_N \setminus H)$ will therefore be purely discrete. There is no difficulty obtaining Weyl's law for this half. Cf. [37, §§6.5, 6.7] and [30, pp.69, 72 (†)].

The Sarnak-Phillips philosophy refers mainly to the *other* (i.e. "even") half of L_2.

Since the first (nonzero) eigenvalue of *any* triangle group is automatically bigger than $\frac{1}{4}$ (cf. [10, p.583 (8)]), the relevant conjecture can be stated as follows:

$$(\bigstar)\qquad \left\{ \begin{array}{l} \text{for } N \neq 3,4,6, \text{ the Hecke group } \mathbf{G}(2\cos\frac{\pi}{N}) \\ \text{should admit } \textit{no} \text{ even cusp forms} \end{array} \right\}.$$

For $N = 3, 4, 6$, the quotient $\mathbf{G}_N \setminus H$ is arithmetic and even cusp forms will exist *in abundance.* In fact, Weyl's law holds exactly as in the case of "odd" R. (cf. [10, pp.511 (top), 476] and equation (4.15) below.)

The contrast between odd/even and arithmetic/nonarithmetic is indeed striking. It is now apparent why $\mathbf{G}_N \setminus H$ is such a natural candidate for some computer experimentation.

§2. The procedure in a nutshell

Our aim is to find cusp forms $\varphi(z)$ for $\mathbf{G}(2\cos\frac{\pi}{N})$. By virtue of an earlier remark, we already know λ must be strictly greater than $\frac{1}{4}$. This leads to $R > 0$ and a Fourier expansion

$$(2.1)\qquad \varphi(x+iy) = \sum_{n=1}^{\infty} c_n y^{\frac{1}{2}} K_{iR}\left(\frac{2\pi n y}{\mathcal{L}}\right) \left\{ \begin{array}{c} \cos\left(\dfrac{2\pi n x}{\mathcal{L}}\right) \\ \text{--------} \\ \sin\left(\dfrac{2\pi n x}{\mathcal{L}}\right) \end{array} \right\}$$

depending on whether φ is even or odd. As usual: $\lambda = \frac{1}{4} + R^2$. There is no loss of generality in assuming that $c_n \in \mathbf{R}$.

The RHS of (2.1) is automatically invariant under $z \mapsto z + \mathcal{L}$. To achieve full automorphy, we need to ensure that

$$(2.2)\qquad \varphi\left(-\frac{1}{z}\right) \equiv \varphi(z).$$

For a "true" cusp form we stress that (2.1) must be absolutely convergent on *all* of H. Cf. [10, pp.23–25].

The K-Bessel function is defined via

$$(2.3)\qquad K_{iR}(X) = \frac{1}{2}\int_{-\infty}^{\infty} e^{-X\cosh t} e^{iRt}\, dt.$$

Due to the extremely small size of $K_{iR}(X)$ for $R > 20$, it is best ([11, 14]) to compute (2.3) by *bending* the contour in a manner similar to stationary phase. In this way: there is no difficulty calculating $\exp(\frac{\pi}{2}R)K_{iR}(X)$ to 10 or 11 places for R-values all the way out to 75000.

The algorithm we use in connection with (2.1) + (2.2) is very similar to [11]. There is *one* major difference, however. Namely: the group $\mathbf{G}_N$ does not generally admit any Hecke operators. Cf. [20], [31, §4], [33]. This effectively *eliminates* any hope of determining R by use of some sort of multiplicative relations among the c_n.

To circumvent this difficulty, we proceed as follows. First of all: recall that (2.2) is equivalent to

$$\sum_{n=1}^{\infty} c_n I_n(z, R) = 0. \;^{3} \tag{2.4}$$

We now select *two* batches of points $\{z_1, \ldots, z_{M-1}\}$ and $\{w_1, \ldots, w_{M-1}\}$ in $\mathcal{F}_N$ (with suitable M) and repeatedly solve

$$\sum_{n=2}^{M} c_n I_n(z_j, R) = -I_1(z_j, R), \quad 1 \leqq j \leqq M-1; \tag{2.5$'$}$$

$$\sum_{n=2}^{M} c_n I_n(w_j, R) = -I_1(w_j, R), \quad 1 \leqq j \leqq M-1. \tag{2.5$''$}$$

The goal is to determine those R-values for which the solution sets $(c_2', c_3', \ldots, c_M')$ and $(c_2'', c_3'', \ldots, c_M'')$ match [as far as possible].

To this end: one simply "checks" (2.5$'$) versus (2.5$''$) on a sufficiently fine R-grid, looking first for approximately coincident (c_k') and (c_k''), and then, in *each* such instance, proceeds to calculate the "point of closest approach" by repeating the comparison on a still finer R-grid.[7]

If the final differences $|c_k' - c_k''|$ are small enough (in an appropriate norm), the resulting R is declared a "success".

If the *same* R-value (and c_k-coefficients) arise for widely disparate z_j & w_j, it is reasonable to expect that one has actually found a true cusp form.

This is the new strategy in a nutshell.

There is very little difficulty modifying the code in [11, appendix A] to accomodate this revised procedure. Cf. [12, appendix A].

These two references also provide further information about the various subtleties that can (and do!) occur.

Far and away the most important thing to worry about is that the **pseudo cusp forms**[8] introduced in [13] are properly *excluded* from occurring in (2.5$'$) (2.5$''$). One does this by requiring that the bulk of the test points z_j, w_j satisfy

$$\mathrm{Im}\; E(z_j) < \mathrm{Im}\,(\rho), \quad \mathrm{Im}\; E(w_j) < \mathrm{Im}\,(\rho), \tag{2.6}$$

[3] And that the terms I_n ultimately decay *exponentially fast* wrt n. ⟦To bring the first few I_n "up" closer to 1, it is customary to premultiply $K_{iR}(X)$ by $\exp(\frac{\pi}{2}R)$. We *tacitly* assume that this has been done. See [4] for the relevant asymptotics of $K_{iR}(X)$.⟧

[7] In line with [11], we'll denote the respective R-increments by *H2* and *H3*.

[8] (and their relatives $a_1 G_s(z\,;\,\tau_1) + \cdots + a_m G_s(z\,;\,\tau_m)$!!)

where $\rho \equiv \exp(\pi i/N)$. (Note that ρ is simply the lower right-hand corner of $\mathcal{F}_N$.)[9]

The final code is implemented in standard Cray-Fortran and is about 1200 lines long. Only single-precision variables are used ...

In solving (2.5), we employ standard Gauss-elimination. Cf. [6, pp.65–72 (II)].

§3. Some theoretical concerns and related caveats

A little thought shows that the algorithm in §2 actually rests on a number of presumptions (whose validity may be troublesome to demonstrate apriori).

First of all, in considering (2.5), it is clear that we have assumed that:

$$c_1 = 1. \tag{3.1}$$

Cusp forms satisfying this condition will be referred to as "unit normalized." Insofar as we are not dealing with any kind of multiple eigenvalue, this normalization seems perfectly legitimate. Things happen for reasons; it is difficult to imagine what $c_1 = 0$ could possibly *mean* (especially in a nonarithmetic setting).

It is essential to bear in mind here that $\mathbf{G}_N$ is a *maximal* Fuchsian group. Cf. [7, 22, 34]. As such: its normalizer can't be something strictly bigger. This effectively rules out any kind of *intrinsic* (or representation-theoretic) reason for multiple eigenvalues.[10] Compare [25].

In our actual experiments, R will be kept less than 60 or so, while N will be taken $\leqq 7$.

The hope (in each instance) will be to attain 6 decimal place accuracy for the R_n.

In line with this, the parameters $H2$ and $H3$ will be taken to be 10^{-3} and 10^{-6}, respectively.

To the extent that Weyl's law *does* hold, the average distance between successive odd (or even) R_n will be about $4\pi/AR$, where

$$A = \pi\left(1 - \frac{2}{N}\right) = \text{the hyperbolic area of } \mathcal{F}_N.$$

[9] In connection with footnote 5, we stress that the $\mathbf{G}_N$-orbit of ρ never rises above $y = \sin(\pi/N)$. Cf. [20, p.201 (middle)].

[10] (the relevant format being simply $\varphi(\gamma z)$)

To minimize any potential difficulties caused by multiple, *or nearly multiple*, eigenvalues, one needs to insist that $H2$ be a tiny fraction of $4\pi/AR$. For the above-mentioned values, one easily checks that:

$$H2 \leqq \left\{ \begin{matrix} 0.50\,\% \\ 1.00\,\% \\ 1.50\,\% \end{matrix} \right\} \quad \text{of} \quad \frac{4\pi}{AR} \quad \text{when} \quad \left\{ \begin{matrix} R \leqq 20 \\ R \leqq 40 \\ R \leqq 60 \end{matrix} \right\}.$$

Beyond *hoping* that these percentages are sufficient, it is also apparent that the success of our algorithm will hinge (just as importantly!) on how rapidly the solutions of (2.5′) (2.5″) vary wrt R.

Indeed, even in the best of cases, one has to contend with the possible occurrence of (small) regions of ill-conditioning for one or both systems. This issue becomes increasingly important as the size of R & M grows, and is one of the main reasons for our insisting that a variety of (disparate) batches $\{z_j\} \cup \{w_j\}$ be used.

With each of these concerns, there are simply *no* apriori guarantees.[11]

This lack of guarantees is balanced, however, by the fact that R is quite modest. The total number of eigenvalues involved here is simply not that great. If suspicious results do occur, one is always free to test another batch $\{z_j\} \cup \{w_j\}$, or to reduce $H2$ & $H3$.

This point-of-view needs to be kept in mind when considering Tables 1,2,3,6,7. Though we lack any kind of rigorous proof of completeness, the reasonableness of our parameters [and built-in stability checks!] will *tend* to give us confidence that nothing has been missed.[12]

In the future, some way of eliminating most (*or* all) of the guesswork in (2.5) may yet be found.

In the meantime, one fact to keep in mind is that the modified Laplacians considered in [39, p.196] all have compact resolvent (with appropriately smooth dependence on the cut-off level a). It is therefore conceivable that our approach ⟦using (2.5)⟧ may somehow be *combined* with that of [39, 40] in order to create a "hybrid" method characterized by a significantly higher level of both rigor and computational control.

See [5] for some ideas in this direction (after taking due note of the last 4 paragraphs of §6).

[11] Part of these difficulties are, of course, common to any eigenvalue computation.

[12] Any residual uncertainties here are best viewed as reflections of the fact that this whole area remains largely in an exploratory phase.

§4. Coefficient relations for $N = 4$ and 6

Since $\mathbf{G}_4$ and $\mathbf{G}_6$ are commensurable with $\mathrm{PSL}(2, \mathbf{Z})$, it is reasonable to conjecture that *some* type of Hecke operators will now exist — and that the Fourier coefficients of any unit-normalized φ automatically satisfy certain multiplicative relations (at least if λ has multiplicity 1).

To treat $N = 4$ and 6 simultaneously, we set

$$q = \frac{N}{2}$$

and (then) *observe* that $\mathcal{L} = 2\cos(\frac{\pi}{N}) = \sqrt{q}$.

Let $\mathcal{G}$ be the subgroup of $\mathrm{PSL}(2, \mathbf{R})$ which is generated by

$$w \mapsto w + 1 \quad \text{and} \quad w \mapsto -\frac{1}{qw}.$$

The group $\mathcal{G}$ is nothing but $\mathbf{G}_N$ viewed under the auxiliary mapping

$$z = \sqrt{q}w. \tag{4.1}$$

We also set:

$$\Gamma_0(q) = \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathrm{SL}(2, \mathbf{Z}); c \equiv 0 \bmod q \right\}.$$

The discussion in [17] shows that $\Gamma_0(q) \leqq \mathcal{G}$ and that the index is 2. We already know that $\mathcal{G}$ is a *maximal* Fuchsian group. Cf. §3. The analysis in [1, p.139] immediately implies that:

$$\mathcal{G} = \text{ the normalizer of } \Gamma_0(q).$$

With these items in place, it is now possible to derive an important connection between cusp forms on $\mathrm{PSL}(2, \mathbf{Z})$, $\mathbf{G}_N$, and $\Gamma_0(q)$.

To explain things, we assume that the reader already has *some* familiarity with the Atkin-Lehner formalism [1] *and* is willing to grant that similar things should hold for nonholomorphic cusp forms. Compare [21, 26].

In the remarks that follow, we restrict ourselves to the case of "even" forms. The "odd" case is similar.

To get started: let $f_0(z)$ be any Hecke-normalized cusp form on $\mathrm{PSL}(2, \mathbf{Z})$ with eigenvalue $\lambda \equiv \frac{1}{4} + R^2$. We therefore have

$$f_0(x + iy) = \sum_{n=1}^{\infty} a_n y^{\frac{1}{2}} K_{iR}(2\pi n y) \cos(2\pi n x) \tag{4.2}$$

and

$$(4.3) \qquad \sum_{n=1}^{\infty} \frac{a_n}{n^s} = \prod_p \frac{1}{1 - a_p p^{-s} + p^{-2s}}.$$

Cf. [11, equation (2.3)].

By an old remark of Rausenberger [27], the function

$$(4.4) \qquad g_0(z) = f_0(z\sqrt{q}) + f_0\left(\frac{z}{\sqrt{q}}\right)$$

is automorphic (hence *cuspidal*) on $\mathbf{G}_N$. A trivial manipulation yields:

$$(4.5) \qquad g_0(z) = \left(\frac{1}{\sqrt{q}}\right)^{1/2} \sum_{n=1}^{\infty} c_n y^{\frac{1}{2}} K_{iR}\left(\frac{2\pi n y}{\sqrt{q}}\right) \cos\left(\frac{2\pi n x}{\sqrt{q}}\right)$$

where

$$(4.6) \qquad c_n = a_n + \sqrt{q} a_{\frac{n}{q}}.$$

The symbol $a_{n/q}$ is understood to be 0 if $q \nmid n$.

By applying $z = \sqrt{q}w$, we see that

$$(4.7) \qquad h_0(w) = f_0(qw) + f_0(w)$$

is cuspidal on $\mathcal{G}$. Since f_0 "lives" on $\mathrm{PSL}(2, \mathbf{Z})$, the function h_0 is an *old*-form on $\Gamma_0(q)$. Cf. [1, pp.145–146].

The Fourier expansion of $h_0(w)$ is simply

$$(4.8) \qquad h_0(u + iv) = \sum_{n=1}^{\infty} c_n v^{\frac{1}{2}} K_{iR}(2\pi n v) \cos(2\pi n u).$$

This expansion is augmented by the relation:

$$(4.9) \qquad h_0\left(-\frac{1}{qw}\right) = h_0(w).$$

We now turn matters completely around and *begin* with any *new*-form $h(w)$ on $\Gamma_0(q) \setminus H$. Cf. [1, p.145] and [26, pp.321–328].

There are two types of new-forms depending on whether

$$h\left(-\frac{1}{qw}\right) = \pm h(w).$$

Cf. [1, p.147]. To obtain automorphy on $\mathcal{G}$, we obviously want the "+" sign to hold. Such new-forms will be called "proper." (The underlying theme here is essentially one of *invariant* subspaces. Cf. [1, Lemma 25].)

For proper h, we have:

$$h(w) = \sum_{n=1}^{\infty} c_n v^{\frac{1}{2}} K_{iR}(2\pi n v) \cos(2\pi n u) \tag{4.10}$$

and

$$\sum_{n=1}^{\infty} \frac{c_n}{n^s} = \frac{1}{1 - c_q q^{-s}} \prod_{p \neq q} \frac{1}{1 - c_p p^{-s} + p^{-2s}} \, ; \tag{4.11$_\mathrm{A}$}$$

$$c_q = -\frac{1}{\sqrt{q}}. \tag{4.11$_\mathrm{B}$}$$

Cf. [1, pp.147, 150].

By applying $z = \sqrt{q}w$, we see that

$$g(z) = h(\frac{z}{\sqrt{q}}) \tag{4.12}$$

is cuspidal on $\mathbf{G}_N$ and has Fourier expansion:

$$g(z) = \left(\frac{1}{\sqrt{q}}\right)^{1/2} \sum_{n=1}^{\infty} c_n y^{\frac{1}{2}} K_{iR}\left(\frac{2\pi n y}{\sqrt{q}}\right) \cos\left(\frac{2\pi n x}{\sqrt{q}}\right). \tag{4.13}$$

It makes sense to call g a *new*-form on $\mathbf{G}_N$. The earlier function g_0 will (then) be called an *old*-form.

By abuse of language, we can use the same terminology for cg and cg_0, $c \neq 0$.

By reviewing the definition of new-form in [1, 26], it is easily seen that any two g and g_0 must be orthogonal (independent of their eigenvalue). In fact:

$$\langle g, g_0 \rangle = \langle h, h_0 \rangle_{\mathcal{G}} = \frac{1}{[\mathcal{G} : \Gamma_0(q)]} \langle h, h_0 \rangle_{\Gamma_0(q)} = 0.$$

This observation leads to the following reformulation. Let $\lambda = \frac{1}{4} + R^2$ be any eigenvalue for $\mathbf{G}_N \setminus H$ *with multiplicity* 1. Let the corresponding unit-normalized eigenfunction be φ.

If λ is an eigenvalue for $\mathrm{PSL}(2,\mathbf{Z})$, then

$$\varphi = (\sqrt{q})^{1/2} g_0 \tag{4.14$_\mathrm{A}$}$$

for a uniquely determined f_0 on $\mathrm{PSL}(2,\mathbf{Z}) \setminus H$. Otherwise,

$$\varphi = (\sqrt{q})^{1/2} g(z) \tag{4.14$_\mathrm{B}$}$$

for a uniquely determined (proper) new-form h on $\Gamma_0(q) \setminus H$.

All that we're *really* doing here is looking at the old-form/new-form decomposition of $\varphi(w\sqrt{q})$ on $\Gamma_0(q) \setminus H$ ...

We won't worry about multiplicity > 1.

Watching *old*-forms appear [on the machine] and verifying[13] the *relations* implicit in (4.3), (4.6), (4.11) should prove quite interesting. Especially: $(4.11)_\mathrm{B}$.

We summarize things with a diagram:

normalized f_0 on $PSL(2,\mathbf{Z})$		old φ on $\mathbf{G}_N\backslash H$
coefficients a_n	$\longleftrightarrow$	coefficients $c_n = a_n + \sqrt{q}a_{n/q}$

proper newform h on $\Gamma_0(q)$		new φ on $\mathbf{G}_N\backslash H$
coefficients c_n	$\longleftrightarrow$	coefficients c_n

Incidentally: in the case of old forms, one naturally expects that $\{f_0(w), f_0(qw)\}$ will be a *basis* for the corresponding eigenspace of $\Gamma_0(q)$. The associated multiplicity on $\Gamma_0(q)$ will therefore by 2. [On $\mathcal{G}$, it'll be *1*.]

The corresponding numerology with regard to Weyl's law then goes as follows:

$$\begin{tikzcd}
\mathrm{PSL}(2,\mathbf{Z}) & & \\
& \Gamma_0(q) \arrow[ul, "q+1"'] \arrow[dl, "2"] & \\
\mathcal{G} & &
\end{tikzcd}
\qquad
\begin{array}{l}
\mu = \mu(\mathcal{F}_3) = \frac{\pi}{3} \\[4em]
A = \mu(\mathcal{F}_N) = \pi(1 - q^{-1})
\end{array}$$

[13] for both old and new!

$$p = N[\text{proper new-forms with } \lambda_n \leqq x]$$

$$i = N[\text{improper new-forms with } \lambda_n \leqq x]$$

$$\frac{(q+1)\mu}{4\pi} = (\text{multiplicity } 2)\frac{\mu}{4\pi}X + p + i$$

$$\frac{A}{4\pi}X = \frac{\mu}{4\pi}X + p$$

$$\frac{A}{4\pi}X = \frac{\mu}{4\pi}X + i \quad [\text{by switching } + \text{ to } - \text{ in eq. (4.4)}].$$

Since $(q+1)\mu = 2A$, everything is consistent, and we simply find that:

$$p = \frac{1}{2}(q-1)\frac{\mu}{4\pi}X + [\text{lower order terms}]$$

$$i = \frac{1}{2}(q-1)\frac{\mu}{4\pi}X + [\text{lower order terms}].$$

Before closing this section, we stop to point out a useful fact concerning Eisenstein series.

Let $E_N(z\,;\,s)$ be the obvious Eisenstein series for $\mathbf{G}_N \setminus H$ with $N = 4, 6$. Cf. [10, pp.569, 280] for the proper normalization. The analysis near (4.4) is easily modified to show that

$$\text{(4.15)} \qquad \left\{ \begin{aligned} E_N(z\,;\,s) &= \frac{1}{1+q^s}\left[E_3(z\sqrt{q}\,;\,s) + E_3\left(\frac{z}{\sqrt{q}}\,;\,s\right)\right] \\ \varphi_N(s) &= \varphi_3(s)\frac{1+q^{1-s}}{1+q^s} \end{aligned} \right\}$$

where E_3 and φ_3 correspond to $\mathrm{PSL}(2,\mathbf{Z}) \setminus H$.

These relations reflect the fact that $\mathbf{G}_4$ and $\mathbf{G}_6$ are both arithmetic.

§5. "Odd" eigenvalues for $N = 4{,}5{,}6$

Prior to giving the results, it is useful to say just a few words about the procedure.

Our primary goal [in this set of experiments] was to compute the odd eigenvalues of $\mathbf{G}_4, \mathbf{G}_5, \mathbf{G}_6$ with $R \leqq 25$ to an R-accuracy of *six* decimal places.

One wished to do this as efficaciously as possible — which basically meant that (2.5) had to be "optimally conditioned." This, in turn, meant that some caution had to be exercised in the choice of z_j and w_j.

A batch $\{z_j\} \cup \{w_j\}$ is said to be of *type* $(\alpha_1, \ldots, \alpha_r \| \beta_1, \ldots, \beta_s)$ when:

(i) the points $E(z_j)$ are distributed in some regular fashion along the line segments $\{0 \leqq x \leqq \frac{1}{2}\mathcal{L},\, y = \alpha_i\}$, $1 \leqq i \leqq r$;

(ii) similarly for $E(w_j)$ and $\{0 \leqq x \leqq \frac{1}{2}\mathcal{L},\, y = \beta_k\}$, $1 \leqq k \leqq s$.

To discourage pseudo cusp forms (as in §2), one requires that:

$$\alpha_i < \sin\left(\frac{\pi}{N}\right), \qquad \beta_k < \sin\left(\frac{\pi}{N}\right).$$

Since (2.4) must hold at *any* $z \in H$, it is *not* necessary that the (original) points z_j and w_j lie in $\mathcal{F}_N$. Indeed: for purposes of achieving better conditioning, it would seem *wise* to let $E(z_j)$ and $E(w_j)$ range all the way out to $x = \frac{1}{2}\mathcal{L}$. (Intuitively: one wants to spread things out a bit. Cf. Figure 1. Several test runs with $N = 6$ convinced us early on that this "trick" would be a very good idea. We adopted it without further ado.)

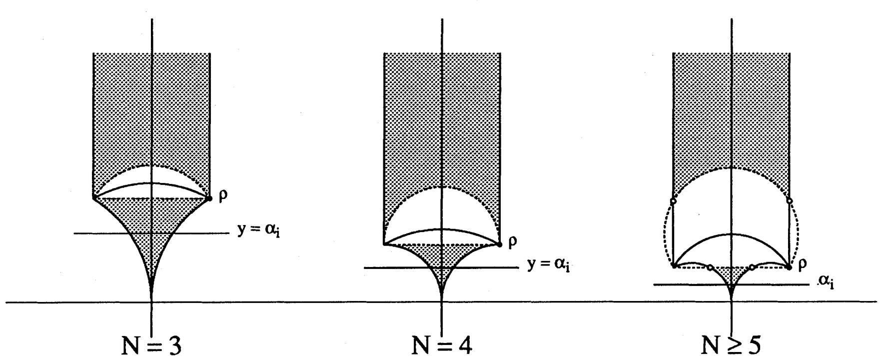

Figure 1

Any number of *other* configurations were (actually) tested before we finally settled on type $(\alpha\|\beta)$. One curious finding was that the *vertical* batches used in [11] do *not* seem to condition so well once N starts to increase.

Our production jobs were all of type $(\alpha_1, \alpha_2\|\beta_1, \beta_2)$. The parameters were as follows:

$N = 4$	$N = 5$	$N = 6$
$(.60, .65 \| .62, .67)$	$(.45, .50 \| .47, .52)$	$(.40, .45 \| .42, .47)$
$(.50, .55 \| .52, .57)$	$(.40, .45 \| .42, .47)$	$(.35, .40 \| .37, .42)$
$\sin\frac{\pi}{4} = .70711$	$\sin\frac{\pi}{5} = .58779$	$\sin\frac{\pi}{6} = .50000$
$H2 = .001$; $H3 = 10^{-6}$		

As noted earlier, the algorithm outlined in §2 was implemented in standard CRAY-Fortran. In doing so: we were especially careful to arrange things so that, by deleting several z_j and w_j, it would[14] be possible to treat *several* M-values in parallel — at least up to those points where (2.5) actually needed to be solved. [This is done by appropriately structuring the "flow pattern" through levels $H2$ and $H3$.]

For safety: we (then) worked with 3 such M-values in our actual runs. Since the number of distinct $(\alpha\|\beta)$ types is 2, this gives an effective total of 6 "tracks."

The choice of M changes with R. This is necessary to ensure "admissibility" in the sense of [11, eq.(2.6)]. That is: we need to have

$$(5.1) \qquad |I_\ell(z_j, R)| \leqq \begin{pmatrix} \text{something like} \\ 10^{-9} \end{pmatrix} \cdot \max_{\substack{1 \leqq k \leqq M-1 \\ 1 \leqq n \leqq M}} |I_n(z_k, R)|$$

for $\ell > M$.

It is not wise to overshoot by too much on this aspect of the code.

The variability in M means that the grid points z_j and w_j must also be (occasionally) changed as well.

Fortunately: these changes are all very gradual.

[14](in effect)

For *each* $(\alpha\|\beta)$ type, we then prepared a list of semifinal R-values by scanning the 3 outputs (wrt M) for the best $|c'_k - c''_k|$ values.

To obtain the final R-values, we repeated this procedure (on the *2* semi-final lists).

The difference between the semifinal and final R-values was usually less than $\frac{1}{2} \times 10^{-6}$ and never greater than 1×10^{-6}.

Our final R_n-values are shown in Tables 1–3.

$N = 4$

7.220872	16.138073*	20.530160	24.028513
9.533695*	16.644259*	21.049526	24.419715*
11.317680	17.493113	21.479057*	25.050855*
12.173008*	18.180918*	22.194674*	25.119336
13.310164	18.437078	22.374933	...
14.358510*	19.484714*	23.201396*	
15.274023	20.106695*	23.263712*	
Odd Eigenvalues for $\mathbf{G}(2\cos\frac{\pi}{4})$			
* indicates an old-form			

Table 1

$N = 5$

6.473700	15.176893	19.385430	23.052526
8.636765	15.759928	19.962241	23.438611
10.136450	16.276410	20.597938	23.509476
11.015570	16.890976	20.745577	24.001860
12.084067	17.757303	21.287052	24.239718
12.851289	18.031441	21.675649	24.631401
14.071834	18.633434	22.197638	25.081315
14.307857	19.011695	22.399384	...
Odd Eigenvalues for $\mathbf{G}(2\cos\frac{\pi}{5})$			

Table 2

$N = 6$

6.120576	15.483162	20.106695*	23.460177
8.193036	16.138073*	20.409439	24.209622
9.533695*	16.644259*	21.108696	24.419715*
10.507607	16.965398	21.479057*	24.916657
11.365904	17.820675	21.612650	24.952648
12.173008*	18.018977	22.100313	25.050855*
13.378621	18.180918*	22.194674*	...
13.507911	19.026777	22.671118	
14.358510*	19.484714*	23.201396*	
14.787325	19.566910	23.263712*	
Odd Eigenvalues for $\mathbf{G}(2\cos\frac{\pi}{6})$			
* indicates an old-form			

Table 3

Tables 4 and 5 supply some additional data. For information about CPU times, see §8(B).

N	type	$R \approx 10$	$R \approx 18$	$R \approx 25$
4	$(.60, .65 \,\|\, .62, .67)$	12	16	19
4	$(.50, .55 \,\|\, .52, .57)$	15	18	22
5	$(.45, .50 \,\|\, .47, .52)$	18	23	27
5	$(.40, .45 \,\|\, .42, .47)$	21	25	30
6	$(.40, .45 \,\|\, .42, .47)$	22	27	32
6	$(.35, .40 \,\|\, .37, .42)$	25	31	36
Sample M-values				

Table 4

N	R-range	2	3	4	5	6	7	8	9	10	12	14	16
4	$10 \sim 15$	E-6	E-6	E-6	E-6	E-6	E-6	E-5	E-5	E-4	E-2	*	*
4	$20 \sim 25$	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-5	E-4	E-3	E-2
5	$10 \sim 15$	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-5	E-5	E-3	E-2	*
5	$20 \sim 25$	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-5	E-5
6	$10 \sim 15$	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-5	E-5	E-4	E-3	E-2
6	$20 \sim 25$	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-6	E-5	E-5
Typical orders-of-magnitude for $\lvert c'_k - c''_k \rvert$													
using semifinal R_n-values													
N. B. The *best* cases are typically better by 1 or 2 orders.													

Table 5

Thus far we have emphasized R_n and *not* the associated Fourier coefficients c_k. Though the latter are certainly of interest, there seems to be very little point in making complete lists of all the Fourier coefficients that were actually obtained.

In §7, we'll discuss 15 typical (*or* otherwise interesting!!) examples in greater detail.

Readers needing more information than this are advised to contact the author; the complete mass of Fourier coefficients is available on magnetic tape.

In scanning this output, we discovered *no* counterexamples to the Ramanujan-Petersson conjecture (for $N = 4, 6$). Cf. [18] and equations (4.3), (4.11).

§6. "Even" eigenvalues for N = 4,5,6,7

The situation for the arithmetic cases $N = 4$ and 6 is very similar to §5. Our *aim* was to compute the even eigenvalues ⟦of $\mathbf{G}_4$ and $\mathbf{G}_6$⟧ with $R \leqq 25$ to an R-accuracy of six decimal places.

Our production jobs were all of type $(\alpha_1, \alpha_2 \| \beta_1, \beta_2)$. The parameters were as follows:

$N = 4$	$N = 6$
$(.50, .55 \,\|\, .52, .57)$	$(.40, .45 \,\|\, .42, .47)$
$(.40, .45 \,\|\, .42, .47)$	$(.35, .40 \,\|\, .37, .42)$
$\sin \frac{\pi}{4} = .70711$	$\sin \frac{\pi}{6} = .50000$
$H2 = .001$;	$H3 = 10^{-6}$

In an attempt to gain better accuracy, we decided to run things using *six* M-values (instead of just 3).

Our final R_n-values [for $N = 4, 6$] are displayed in Tables 6 and 7.

$N = 4$

8.922877	17.878003	22.785909*
10.920392	19.125423	23.496586
13.779751*	19.423481*	24.112353*
14.685016	20.547604	24.856199
16.404109	21.315796*	25.052424
17.738563*	22.089045	...
Even Eigenvalues for $\mathbf{G}(2\cos\frac{\pi}{4})$		
* indicates an old-form		

Table 6

$N = 6$

5.098742	16.736215	21.807127
8.038861	17.500557	22.659272
9.743749	17.738563*	22.785908*
11.346418	18.647430	22.839291
11.889976	18.962642	23.620927
13.135144	19.423482*	23.979851
13.779751*	19.896104	24.112353*
14.626236	20.664907	24.298256
15.799494	21.315796*	24.931087
16.270959	21.434643	...
Even Eigenvalues for $\mathbf{G}(2\cos\frac{\pi}{6})$		
* indicates an old-form		

Table 7

The discrepancy between the semifinal and final R-values was, except for the last 4 entries in Table 7, entirely similar to §5. [The exceptions seemed to be caused by a conditioning problem with one of the types. The *other* type worked perfectly fine ...]

Tables 8 and 9 (top) supply some additional data.

See §8 (B) for information about CPU times and §7 for various examples illustrating the actual Fourier coefficients.

⟦No counterexamples to Ramanujan-Petersson were found in a scan through the total mass of computed c_k ...⟧

The *focus* in the *non*arithmetic cases was (of course) quite different.

Here one basically wished to investigate conjecture (★). We attacked this problem in the following ranges:

$$\left\{ \begin{array}{c} 0 \leqq R \leqq 60 \quad \text{for } N = 5 \\ \hdashline 0 \leqq R \leqq 40 \quad \text{for } N = 7 \end{array} \right\}.$$

In both cases: **no even cusp forms were found**.

This assertion requires some elaboration, however.

Up to a point: the basic procedure is exactly like before. Our production jobs had the following parameters

$N = 5$	$N = 7$
$(.50, .55 \,\|\, .52, .57)$ $(.40, .45 \,\|\, .42, .47)$ $(.30, .35 \,\|\, .32, .37)$ for $R < 15$	$(.30, .35 \,\|\, .32, .37)$
$(.45, .50 \,\|\, .47, .52)$ $(.40, .45 \,\|\, .42, .47)$ for $R > 15$	
$\sin \frac{\pi}{5} = .58779$	$\sin \frac{\pi}{7} = .43388$
$H2 = .001$;	$H3 = 10^{-6}$

Σ

For $N = 5$ and $0 \leqq R \leqq 15$, we ran 6 (and sometimes 9) M-values in parallel. In all other cases: we used 3.

The case $N = 7$ was pursued in earnest only *after* $N = 5$ was complete. Its purpose was (thus) mainly one of insurance. To save computer time, we decided to proceed with only one "type."

In §3, we explained why it is important to keep the numbers α_i and β_k below $\sin(\frac{\pi}{N})$. Placing these levels *too low*, however, causes M to become rather large — which begins to affect the overall accuracy (and CPU time) adversely. It is therefore necessary to strike some kind of balance.

One might think that running jobs with the parameters shown above would simply produce no output [in accordance with ($\bigstar$)]. This, however, is *not* the case.

The assertion that "no even cusp forms were found" is *not* as simple as it looks.

What typically happened in our (even) nonarithmetic jobs was that R-values would occasionally come out showing differences $|c_k' - c_k''|$ that looked "half-way" respectable.

BUT (and this is the key point!): *one or more* danger signs would *invariably* apply.

These signs included:

(a) excessive movement (or disappearance!) of the proposed R-value when M is varied;

(b) excessive movement (or disappearance!) of the proposed R-value when the "type" is varied;

(c) excessive movement in the first few Fourier coefficients under similar variations;
(d) values of $|c'_k - c''_k|$ that were typically 3 to 4 orders-of-magnitude *worse* than their "counterparts" for *odd* R (in the same range).

Item (d), on its own, was usually enough to destroy any putative R.

The essential point here is that [philosophically] one should expect similar levels of "stability" to be exhibited by both the even and odd R-values. This was certainly the case for $N = 4, 6$.[15]

One always has to be a bit careful in situations like this to exclude the possibility that some type of intrinsic "static" region (wrt R) is the real culprit in (a)–(d). On such R-regions, there could easily be an overall degradation in conditioning-level [which causes (a)–(d)].

This effect was discussed in [11] at some length. The same effect certainly occurs for $N = 4, 5, 6, 7$. In fact: here it begins even earlier [with "missing" R_n-values occurring on one-or-another track for R *as low* as 9.533].

At this stage of the game, one can only *hope* that the (potential) effects of such "static" are indeed minimized by running several tracks and/or types.[16]

In this connection: the most interesting thing is that some R-values with half-way decent $|c'_k - c''_k|$ *did* manage to stay reasonably "intact" [wrt (a)–(c)] under *several* changes of track.

For $N = 5$ at least, such R-values seemed to be most common [and "strongest"] in cases where α and β could come closest to $\sin(\frac{\pi}{N})$.

In view of this $\alpha\beta$-dependence, it is natural to *conjecture* that the foregoing R-values must simply represent *some type* of "residual" effect from the pseudo cusp forms mentioned in §2.

This issue is carefully explored in [12] — on both the theoretical and experimental fronts. The proposed explanation [in terms of pseudo-residuals] is found to be very well-substantiated by further experimentation.

The upshot of these remarks is very simple. The occurrence of "pseudo-residuals" makes it *doubly* important to pay close attention to (a)–(d) [*and to* the size of α_i and β_k]. Failure to do so may cause one to "snare" the wrong type of "animal" altogether ...

---

[15]Though this expectation seems reasonable enough for arbitrary N, we make *no* pretenses about having a rigorous proof. The full statement for $N = 5, 7$ should therefore read: no even cusp forms *meeting a reasonable set of standards* were detected by the machine.

[16](the rationale being that such "static" regions should occur *randomly* and with *small* relative measure)

Tables 8 and 9 provide some additional information about our "even" runs. Compare: Tables 4,5 in §5.

N	R-range	2	3	4	5	6	7	8	9	10	12	14	16
4	$10 \sim 15$	E-7	E-7	E-7	E-6	E-6	E-6	E-6	E-6	E-4	E-2	*	*
4	$20 \sim 25$	E-6	E-6	E-6	E-6	E-6	E-6	E-5	E-5	E-5	E-5	E-4	E-2
6	$10 \sim 15$	E-7	E-7	E-7	E-7	E-6	E-6	E-6	E-5	E-5	E-3	E-1	*
6	$20 \sim 25$	E-7	E-7	E-7	E-7	E-7	E-6	E-6	E-6	E-6	E-5	E-5	E-5
Typical orders-of-magnitude for $\lvert c_k' - c_k'' \rvert$													
using semifinal R_n-values													
N. B. The *best* cases are typically better by 1 or 2 orders.													

Table 8

N	type	$R \approx 10$	$R \approx 18$	$R \approx 25$	$R \approx 40$	$R \approx 50$	$R \approx 60$
4	$(.50, .55 \parallel .52, .57)$	15	18	22	$*$	$*$	$*$
4	$(.40, .45 \parallel .42, .47)$	17	21	25	$*$	$*$	$*$
6	$(.40, .45 \parallel .42, .47)$	21	26	31	$*$	$*$	$*$
6	$(.35, .40 \parallel .37, .42)$	23	29	34	$*$	$*$	$*$
5	$(.45, .50 \parallel .47, .52)$	$*$	23	26	40	46	52
5	$(.40, .45 \parallel .42, .47)$	21	25	29	44	51	58
7	$(.30, .35 \parallel .32, .37)$	34	40	48	64	$*$	$*$
Sample M-values							

Table 9

To round things out, we now compare our results with those of Winkler.

On the matter of (★), there is little to comment on beyond reporting that Winkler tested $N = 5, 7, 8$ for $R \leqq 20$ and found *no* even cusp forms.

For $N = 4$ and 6, Winkler's results go as follows:

$N = 4$
8.922877
10.920392
14.685016
16.40411
19.12512
even R

Table 10

$N = 6$
5.098742
8.038861
9.743749
11.34642
11.889976
13.135144
14.626227
15.799498
16.736246
17.500559
even R

Table 11

For the most part, Winkler's entries show excellent agreement with our values.[17] Much more striking, however, is the fact that *no* old-forms were detected!! (Two new-forms were also missed: 17.878003 for $N = 4$ and 16.270959 for $N = 6$.)

[17] Any discrepancies are easily explained by looking at the control numbers listed in the 2^{nd} column of Winkler's original tables [39, p.200].

With regard to the old-forms ⟦and examples 7,10,11 in §7⟧, we wish to stress that, near such R, our CRAY output was always quite stable (and well-conditioned).[18]

In particular: there is absolutely no indication of any kind of "hidden" *new*-form also being present. (To achieve the proper perspective on this, it is helpful to review (4.11_{A}), [33, p.80 (bottom)], and the remark about $\langle g, g_0\rangle$ in §4.)

Before implementing any kind of "hybrid" technique (as suggested at the end of §3), it would obviously be desirable to learn why Winkler's method missed the values it did [particularly after doing so well on the others].

§7. Some examples

In this section, we'll look at 15 examples which serve to illustrate various aspects of our production runs. The information given in each case will include: appropriately rounded valued of $\frac{1}{2}(c'_k + c''_k)$; a rough indication of $|c'_k - c''_k|$; and a brief description of the "track" used.

When discussing *old*-forms, remember that:

$$c_n = a_n + \sqrt{q}\, a_{\frac{n}{q}}$$

by virtue of (4.6).

Example 1. $R = 7.220872$ ($N = 4$/odd/new-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	−.7071067	2E−8	7	−.0625	5E−4
3	−.9493510	3E−8	8	−.3460	4E−3
4	.5000021	5E−7	9	−.142	3E−2
5	−.869730	6E−6	10	.80	0.1
6	.671435	6E−5	11	−.57	0.3
type $(.60, .65 \,\|\, .62, .67)$			$M = 11$	final R	

[18] See §8(A) for a related example.

As an indication of (overall) accuracy, note that:

$$\frac{1}{\sqrt{2}} = .707106781, \quad |c_2 + \frac{1}{\sqrt{2}}| = .0000001, \quad |c_4 - \frac{1}{2}| = .0000021$$

$$|c_6 - c_2c_3| = .000143 \qquad |c_8 + \frac{1}{2\sqrt{2}}| = .0076$$

$$|c_9 - (c_3^2 - 1)| = .043 \qquad |c_{10} - c_2c_5| = .19\ .$$

Example 2. $R = 12.173008$ ($N = 4$/odd/old-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	a_k
2	1.7034654	1E−8	.2892518
3	−1.2018588	1E−8	−1.2018588
4	−.5072694	4E−9	−.9163332
5	.0395527	6E−11	.0395527
6	−2.0473248	2E−9	−.3476398
7	.4481331	3E−8	.4481331
8	−1.8501922	3E−7	−.5543014
9	.4444580	6E−7	.4444580
10	.06754	6E−5	.01160
11	−.6935	6E−4	−.6935
12	.638	1E−2	1.130
13	−3.11	1.25(!!)	−3.11
type $(.50, .55 \,\|\, .52, .57)$		$M = 17$	final R

As an indication of (overall) accuracy, note that:

$$|a_4 - (a_2^2 - 1)| = .0000002 \qquad |a_6 - a_2a_3| = .0000000$$
$$|a_8 - (a_2^3 - 2a_2)| = .0000015 \qquad |a_9 - (a_3^2 - 1)| = .0000066$$
$$|a_{10} - a_2a_5| = .00016 \qquad |a_{12} - a_3a_4| = .029\ .$$

In view of the (large) difference at $k = 13$, we do *not* take c_{13} seriously.

The situation for $k \geqq 14$ gets progressively worse. In the terminology of [11], we can thus say that the c_n "hump" occurs at about $13 \sim 14$.

Incidentally: observe that

$$a_2 = \begin{Bmatrix} .2892518 & \text{here} \\ .289252 & \text{in [11]} \end{Bmatrix} \qquad a_3 = \begin{Bmatrix} -1.2018588 & \text{here} \\ -1.201858 & \text{in [11]} \end{Bmatrix}$$

$$a_5 = \begin{Bmatrix} .0395527 & \text{here} \\ .042 & \text{in [11]} \end{Bmatrix} \qquad a_7 = \begin{Bmatrix} .4481331 & \text{here} \\ *** & \text{in [11]} \end{Bmatrix}$$

The current a_k-listing is (thus) significantly better than the one in [11]. This improvement basically reflects the *change* in geometry.

Example 3. $R = 24.028513$ ($N = 4$/odd/new-form).

k	$\frac{1}{2}(c_k' + c_k'')$	rough diff.	k	$\frac{1}{2}(c_k' + c_k'')$	rough diff.
2	−.7071066	3E−8	11	−.879083	1E−6
3	.5772141	2E−8	12	.28860	5E−5
4	.5000000	4E−8	13	1.6529	2E−4
5	.2392995	7E−8	14	.0854	3E−4
6	−.4081516	5E−8	15	.143	3E−3
7	−.1212567	2E−7	16	.247	8E−4
8	−.3535511	3E−7	17	−1.09	0.17
9	−.666826	2E−6	18	.68	0.45
10	−.169205	4E−6			
type (.50, .55 ‖ .52, .57)			$M = 22$		final R

To indicate the overall accuracy, note that:

$$|c_2 + \frac{1}{\sqrt{2}}| = .0000002 \qquad |c_4 - \frac{1}{2}| = .0000000$$

$$|c_6 - c_2c_3| = .0000003 \qquad |c_8 + \frac{1}{2\sqrt{2}}| = .0000023$$

$$|c_9 - (c_3^2 - 1)| = .000002 \qquad |c_{10} - c_2c_5| = .000005$$

$$|c_{12} - c_3c_4| = .000007 \qquad |c_{14} - c_2c_7| = .0003$$

$$|c_{15} - c_3c_5| = .005 \qquad |c_{16} - \frac{1}{4}| = .003\ .$$

The c_n "hump" occurs at about $n = 18$.

Before moving onward, we need to draw attention to an important fact. By reviewing Examples 1–3, it becomes apparent that the number $|c'_k - c''_k|$ is *not* a true indicator of the actual *error* in c_k. (This is seen by looking at the multiplicative relations.) To be on the safe side, it seems preferable to use something like $\max[2\times 10^{-7}, 5(\text{diff})]$ as the "basic indicator" of fuzz-level.[19]

Example 4. $R = 6.120576$ ($N = 6$/odd/new-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	−.6716156	1E−9	10	−.21872	3E−5
3	−.5773503	4E−10	11	−.68593	3E−5
4	−.5489325	4E−9	12	.3176	8E−4
5	.3256987	2E−9	13	−.3325	8E−4
6	.3877575	6E−8	14	.99	2E−2
7	−1.4557169	4E−8	15	−.17	2E−2
8	1.040288	1E−6	16	.30	0.64
9	.333334	1E−6			
type (.40, .45 ‖ .42, .47)			$M = 20$		final R

To indicate the overall accuracy, note that:

$$\frac{1}{\sqrt{3}} = .577350269 \qquad |c_3 + \frac{1}{\sqrt{3}}| = .0000000$$

$$|c_4 - (c_2^2 - 1)| = .0000000 \qquad |c_6 - c_2c_3| = .0000000$$

$$|c_8 - (c_2^3 - 2c_2)| = .0000008 \qquad |c_9 - \frac{1}{3}| = .000001$$

$$|c_{10} - c_2c_5| = .00002 \qquad |c_{12} - c_3c_4| = .0007$$

$$|c_{14} - c_2c_7| = .012 \qquad |c_{15} - c_3c_5| = .02\ .$$

[19]To *mollify* purely random effects, (diff) should actually be replaced here by some type of backward average. ⟦Unless $|c'_k - c''_k|$ is abnormally small compared to its neighbors, this modification is usually insignificant ...⟧

The c_n "hump" occurs at about $n = 16$.

Example 5. $R = 12.173008$ ($N = 6$/odd/old-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	a_k
2	.2892519	6E−8	.2892519
3	.5301920	2E−8	−1.2018588
4	−.9163334	3E−8	−.9163334
5	.0395526	6E−8	.0395526
6	.1533590	8E−8	−.3476400
7	.4481331	8E−8	.4481331
8	−.5543028	2E−7	−.5543028
9	−1.6372162	4E−7	.4444643
10	.011442	2E−6	.011442
11	−.691455	7E−6	−.691455
12	−.48581	3E−5	1.10133
13	−.8030	3E−4	−.8030
14	.132	3E−3	.132
15	.02	8E−3	−.05
16	.72	4E−2	.72
17	−.7	0.41	−.7
type (.35, .40 ‖ .37, .42)		$M = 27$	a semifinal R

To indicate the overall accuracy, note that:

$$|a_4 - (a_2^2 - 1)| = .0000001 \qquad |a_6 - a_2a_3| = .0000001$$
$$|a_8 - (a_2^3 - 2a_2)| = .0000003 \qquad |a_9 - (a_3^2 - 1)| = .0000003$$
$$|a_{10} - a_2a_5| = .000001 \qquad |a_{12} - a_3a_4| = .000027$$
$$|a_{14} - a_2a_7| = .002 \qquad |a_{15} - a_3a_5| = .002$$
$$|a_{16} - (a_2^4 - 3a_2^2 + 1)| = .036\ .$$

The c_n "hump" occurs at about $n = 18$.

The current a_k-values are an improvement over those in Example 2.

It is also interesting to compare things with the old result in [10, p.653], [16]:

$R = 12.1730083246797$

$a_2 = .2892518714$	$a_5 = .03955272$
$a_3 = -1.201858761$	$a_7 = .4481$

These numbers were obtained using double-precision arithmetic (and an *un*inspired Newton-Cotes type algorithm for the K-Bessel function).

Example 6. $R = 24.419715$ ($N = 6$/odd/old-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	a_k
2	.9655410	2E−7	.9655410
3	1.0417911	2E−7	−.6902597
4	−.0677319	2E−7	−.0677319
5	1.3158034	2E−7	1.3158034
6	1.0058915	5E−7	−.6664746
7	−.5454961	6E−7	−.5454961
8	−1.0309378	2E−8	−1.0309378
9	−1.719106	2E−6	−.523541
10	1.270463	2E−6	1.270463
11	−.156968	3E−6	−.156968
12	−.070563	3E−6	.046752
13	−1.894287	7E−6	−1.894287
14	−.526697	4E−6	−.526697
15	1.370791	4E−6	−.908247
16	−.927680	5E−6	−.927680
17	.344743	8E−6	.344743
18	−1.659862	1E−5	−.505494
19	−.10064	2E−5	−.10064
20	−.08913	2E−5	−.08913

21	−.56832	6E−5	.37651
22	−.15148	2E−4	−.15148
23	−.70406	4E−5	−.70406
24	−1.07402	7E−6	.71162
25	.7321	2E−3	.7321
26	−1.8307	3E−3	−1.8307
27	.156	2E−2	1.063
28	.028	2E−2	.028
29	.360	4E−2	.360
30	1.27	1E−1	−.93
31	.29	4E−2	.29
32	−.09	4E−1	−.09
type $(.35, .40 \,\|\, .37, .42)$ $M = 37$			a typical final R (*with* some what larger M)

As an indication of (overall) accuracy, note that:

$$|a_4 - (a_2^2 - 1)| = .0000013 \qquad |a_6 - a_2a_3| = .0000006$$
$$|a_8 - (a_2^3 - 2a_2)| = .0000002 \qquad |a_9 - (a_3^2 - 1)| = .0000005$$
$$|a_{10} - a_2a_5| = .000001 \qquad |a_{12} - a_3a_4| = .000001$$
$$|a_{14} - a_2a_7| = .000002 \qquad |a_{15} - a_3a_5| = .000001$$
$$|a_{16} - (a_2^4 - 3a_2^2 + 1)| = .000002 \qquad |a_{18} - a_2a_9| = .000006$$
$$|a_{20} - a_4a_5| = .000008 \qquad |a_{21} - a_3a_7| = .00002$$
$$|a_{22} - a_2a_{11}| = .00008 \qquad |a_{24} - a_3a_8| = .00001$$
$$|a_{25} - (a_5^2 - 1)| = .0008 \qquad |a_{26} - a_2a_{13}| = .0017$$
$$|a_{27} - (a_3^3 - 2a_3)| = .011 \qquad |a_{28} - a_4a_7| = .009$$
$$|a_{30} - a_2a_{15}| = .053 \qquad |a_{32} - (a_2^5 - 4a_2^3 + 3a_2)| = .23\ .$$

The c_n "hump" occurs at about $32 \sim 33$.

By way of comparison to [11], observe that:

$$a_2 = \begin{Bmatrix} .9655410 & \text{here} \\ .965541 & \text{in [11]} \end{Bmatrix} \qquad a_3 = \begin{Bmatrix} -.6902597 \\ -.690260 \end{Bmatrix}$$

$$a_5 = \begin{Bmatrix} 1.3158034 & \text{here} \\ 1.315804 & \text{in [11]} \end{Bmatrix} \qquad a_7 = \begin{Bmatrix} -.5454961 \\ -.545 \end{Bmatrix}$$

Example 7. $R = 13.77975137$ ($N = 4$/even/old-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	a_k
2	2.9635181	3E−9	1.5493045
3	.2468996	3E−9	.2468996
4	3.5913914	9E−8	1.4003440
5	.7370610	9E−7	.7370610
6	.7316926	3E−7	.3825238
7	−.2614212	8E−7	−.2614212
8	2.600645	2E−6	.620260
9	−.939045	2E−6	−.939045
10	2.1825	2E−3	1.1401
11	−.960	7E−3	−.960
12	1.15	0.37	.61
type (.40, .45 ‖ .42, .47)		$M = 19$	final R

For the overall accuracy, note that:

$$|a_4 - (a_2^2 - 1)| = .0000004 \qquad |a_6 - a_2 a_3| = .0000011$$

$$|a_8 - (a_2^3 - 2a_2)| = .000005 \qquad |a_9 - (a_3^2 - 1)| = .000004$$

$$|a_{10} - a_2 a_5| = .0018 \qquad |a_{12} - a_3 a_4| = .26\ .$$

The c_n "hump" occurs at about $n = 12$.

To make a comparison with [11, 35], observe that:

$$R = \left\{ \begin{array}{lc} 13.77975137 & \text{here} \\ 13.77975135189 & \text{in [11, §10]} \\ 13.7797513519 & \text{in [35]} \end{array} \right\}$$

$$a_2 = \left\{ \begin{array}{lc} 1.5493045 & \text{here} \\ 1.54930447794 & \text{in [11, §10]} \\ 1.5493044779 & \text{in [35]} \end{array} \right\} \quad a_3 = \left\{ \begin{array}{l} .2468996 \\ .24689977245 \\ .2468997725 \end{array} \right\}$$

$$a_5 = \left\{ \begin{array}{l} .7370610 \\ .737060383 \\ .7370603853 \end{array} \right\} \qquad a_7 = \left\{ \begin{array}{l} -.2614212 \\ -.261421 \\ -.2614200758 \end{array} \right\}$$

$$a_{11} = \left\{ \begin{array}{c} -.960 \\ *** \\ -.9535646526 \end{array} \right\}$$

Our earlier remark about c_k-error is nicely illustrated at $k = 3$. ⟦Errors of this kind appear to stem mainly from the fact that we chose H3 to be 10^{-6} in all our production runs. Cf. §§3,5,6. A reduction in H3 should yield better accuracy ...⟧

For the sake of completeness, we also recall that [10, p.653], [16] had:

$R = 13.7797513518907$

$a_2 = 1.54930447794$	$a_3 = .24689977245$
$a_5 = .7370604$	$a_7 = -.2614$

Example 8. $R = 17.878003$ ($N = 4$/even/new-form).

k	$\frac{1}{2}(c_k' + c_k'')$	rough diff.	k	$\frac{1}{2}(c_k' + c_k'')$	rough diff.
2	$-.707106798$	7E$-$11	10	-1.2811050	2E$-$6
3	.9825686	4E$-$12	11	$-.1747944$	5E$-$7
4	.50000000	2E$-$9	12	.491474	1E$-$6
5	1.8117563	1E$-$9	13	1.10189	7E$-$5
6	$-.6947809$	4E$-$8	14	$-.0101$	1E$-$3
7	.0082109	3E$-$8	15	1.795	6E$-$3
8	$-.3535540$	4E$-$7	16	.261	2E$-$3
9	$-.0345570$	1E$-$6	17	.43	0.14
type $(.40, .45 \,\|\, .42, .47)$		$M = 21$			final R

To indicate the overall accuracy, note that:

$$|c_2 + \frac{1}{\sqrt{2}}| = .000000017 \qquad |c_4 - \frac{1}{2}| = .0000000$$

$$|c_6 - c_2c_3| = .0000000 \qquad |c_8 + \frac{1}{2\sqrt{2}}| = .0000006$$

$$|c_9 - (c_3^2 - 1)| = .0000019 \qquad |c_{10} - c_2c_5| = .0000002$$

$$|c_{12} - c_3c_4| = .000190 \qquad |c_{14} - c_2c_7| = .0043$$

$$|c_{15} - c_3c_5| = .015 \qquad |c_{16} - \frac{1}{4}| = .01\ .$$

The c_n "hump" occurs at about $17 \sim 18$.

⟦This R-value is one of the new-forms missed by Winkler.⟧

Example 9. $R = 14.626236$ ($N = 6$/even/new-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	.55536214	7E−10	10	−.5291602	9E−7
3	−.57735035	3E−8	11	−.015748	4E−6
4	−.6915731	3E−7	12	.39927	2E−5
5	−.9528186	3E−7	13	1.0297	1E−3
6	−.3206385	3E−7	14	1.017	9E−3
7	1.8374419	3E−7	15	.66	1E−2
8	−.9394356	1E−7	16	−.05	0.17
9	.3333338	3E−7			
type $(.40, .45 \,\|\, .42, .47)$		$M = 22$		final R	

As an indication of overall accuracy, note that:

$$|c_3 + \frac{1}{\sqrt{3}}| = .00000008 \qquad |c_4 - (c_2^2 - 1)| = .0000002$$

$$|c_6 - c_2c_3| = .0000000 \qquad |c_8 - (c_2^3 - 2c_2)| = .0000001$$

$$|c_9 - \frac{1}{3}| = .0000005 \qquad |c_{10} - c_2c_5| = .0000008$$

$$|c_{12} - c_3c_4| = .00001 \qquad |c_{14} - c_2c_7| = .003$$

$$|c_{15} - c_3c_5| = .11 \qquad |c_{16} - (c_2^4 - 3c_2^2 + 1)| = .22\ .$$

The c_n "hump" occurs at about $n = 16$.

Winkler's value was $R = 14.626227$.

Example 10. $R = 13.7797513515$ ($N = 6$/even/old-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	a_k
2	1.549304477	5E−11	1.549304477
3	1.978950582	6E−10	.246899774
4	1.400344368	1E−9	1.400344368
5	.737060386	1E−8	.737060386
6	3.065997001	1E−8	.382522930
7	−.26142006	1E−8	−.26142006
8	.62025531	1E−8	.62025531
9	−.5113975	3E−8	−.9390405
10	1.1419309	9E−8	1.1419309
11	−.9535642	3E−7	−.9535642
12	2.771212	3E−6	.345744
13	.278822	2E−5	.278822
14	−.40516	2E−5	−.40516
15	1.464	4E−3	.187
16	−.46	2E−2	−.46
17	1.17	0.11	1.17
type (.40, .45 ‖ .42, .47)		$M = 20$	one of our strongest final R

To indicate the overall accuracy, note that:

$$|a_4 - (a_2^2 - 1)| = .000000006 \qquad |a_6 - a_2 a_3| = .000000005$$
$$|a_8 - (a_2^3 - 2a_2)| = .00000000 \qquad |a_9 - (a_3^2 - 1)| = .0000000$$
$$|a_{10} - a_2 a_5| = .0000001 \qquad |a_{12} - a_3 a_4| = .000001$$
$$|a_{14} - a_2 a_7| = .00014 \qquad |a_{15} - a_3 a_5| = .005$$
$$|a_{16} - (a_2^4 - 3a_2^2 + 1)| = .02\ .$$

The c_n "hump" occurs at about $17 \sim 18$.

It is also interesting to compare things with [11, 10, 16, 35] as in Example 7.

	here	[35];[10, p.729]	[11, §10]	[10, p.653]
R-13	.7797513515	.77975135189	.77975135189	.77975135189
a_2	1.549304477	1.54930447794	1.54930447794	1.54930447794
a_3	.246899774	.24689977245	.24689977245	.24689977245
a_5	.737060386	.73706038534	.737060383	.7370604
a_7	−.26142006	−.26142007577	−.261421	−.2614
a_{11}	−.9535642	−.95356465262	$***$	$***$
a_{13}	.278822	.27882702916	$***$	$***$
a_{17}	1.17	1.30734171453	$***$	$***$

The present accuracy is very striking because taking H3 $= 10^{-6}$ would ordinarily suggest $7 \sim 7\frac{1}{2}$ decimal places as being the upper limit. [Bear in mind *too* that $M = 20$, and that we are using only single-precision arithmetic ...]

Example 11. $R = 17.73856338$ ($N = 6$/even/old-form).

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	a_k
2	−.76545805	2E−10	−.76545805
3	.75427190	3E−11	−.97777891
4	−.41407396	1E−9	−.41407396
5	−1.01527351	1E−11	−1.01527351
6	−.57736350	2E−9	.74844873
7	1.18082083	7E−9	1.18082083
8	1.08241430	5E−9	1.08241430
9	−1.7375111	4E−8	−.0439483
10	.7771493	2E−8	.7771493
11	−.6204877	1E−7	−.6204877
12	−.312325	3E−6	.404872
13	.265291	2E−5	.265291

14	$-.90386$	2E$-$5	$-.90386$
15	$-.7659$	5E$-$4	$.9926$
16	$-.414$	2E$-$3	$-.414$
17	$-.135$	2E$-$3	$-.135$
18	1.33	2E$-$2	$.034$
19	$.18$	$.09$	$.18$
type $(.40, .45 \,\|\, .42, .47)$		$M = 27$	one of our strongest final R

For the overall accuracy, note that:

$$|a_4 - (a_2^2 - 1)| = .00000001 \qquad |a_6 - a_2 a_3| = .00000001$$

$$|a_8 - (a_2^3 - 2a_2)| = .00000001 \qquad |a_9 - (a_3^2 - 1)| = .0000001$$

$$|a_{10} - a_2 a_5| = .0000000 \qquad |a_{12} - a_3 a_4| = .000001$$

$$|a_{14} - a_2 a_7| = .00001 \qquad |a_{15} - a_3 a_5| = .0001$$

$$|a_{16} - (a_2^4 - 3a_2^2 + 1)| = .000 \qquad |a_{18} - a_2 a_9| = .000\ .$$

The c_n "hump" occurs at about $n = 20$.

A comparison with [10, pp.653, 729] gives:

	here	[10, p.729]	[10, p.653]
R	17.73856338	17.7385633811	17.7385633811
a_2	$-.76545805$	$-.76545806$	$-.7654580566$
a_3	$-.97777891$	$-.9777789$	$-.9777789075$
a_5	-1.01527351	-1.0152735	-1.0152735
a_7	1.18082083	1.1808208	1.1807
a_{11}	$-.6204877$	$-.6204877$	$***$
a_{13}	.265291	.2652887	$***$
a_{17}	$-.135$	$-.1357407$	$***$

The results in the middle column were obtained by H. Stark using the *same* method as in [35].

The present accuracy is again rather striking.

Example 12. $R = 14.0718340$ ($N = 5$/odd). In a *non*arithmetic case like $N = 5$, it is not so clear what will happen. We therefore look at *both* semifinal values.

$R = 14.0718340$			$R = 14.0718335$		
k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	$-.3658834$	3E−9	2	$-.365882$	1E−6
3	$-.5092615$	1E−8	3	$-.509263$	2E−6
4	$.5249263$	3E−7	4	$.524923$	5E−6
5	$.0132187$	2E−7	5	$.013217$	2E−7
6	$.169034$	5E−7	6	$.169027$	4E−6
7	$.509436$	4E−8	7	$.509436$	5E−8
8	1.007591	2E−7	8	1.007585	6E−7
9	$-.477043$	2E−7	9	$-.477040$	1E−6
10	-1.113400	4E−7	10	-1.11347	3E−5
11	$-.178021$	2E−6	11	$-.17791$	7E−5
12	-1.038675	2E−6	12	-1.040	1E−3
13	$-.53924$	2E−5	13	$-.536$	2E−3
14	-1.2025	3E−4	14	-1.25	4E−2
15	$.639$	2E−3	15	$.72$	.08
16	$-.04$	.06	16	-1.5	1.56
left type $(.40, .45 \,\|\, .42, .47)$			$M = 22$	typical semifinal	
right type $(.45, .50 \,\|\, .47, .52)$			$M = 18$	R-values	

The c_n "hump" occurs at about 17 and 15, respectively. The agreement between the 2 columns (of c_n) is consistent with our earlier remark about fuzz-level. In view of Table 5, the right-hand differences are just about average. The ones on the *left* are significantly better. [Though in this example there was some *advantage* to keeping α_i and β_k further away from $\sin(\frac{\pi}{5})$, the overall situation is basically random.]

There are no (obvious) multiplicative relations.

Example 13. $R = 14.307857$ ($N = 5/\text{odd}$). This example illustrates the possibility of *large* c_n occurring when $\mathbf{G}_N$ is non-arithmetic.

$R = 14.3078567$			$R = 14.3078568$		
k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	10.37429	6E−5	2	10.37433	4E−4
3	−3.15814	2E−5	3	−3.15815	1E−4
4	6.954667	2E−6	4	6.95462	1E−4
5	−5.74760	1E−4	5	−5.74765	2E−4
6	−7.46890	3E−5	6	−7.4690	4E−4
7	−7.69329	8E−5	7	−7.6933	3E−4
8	3.48202	3E−5	8	3.4819	1E−4
9	−6.6204	1E−4	9	−6.6203	3E−6
10	2.5325	2E−4	10	2.531	3E−3
11	.2486	8E−4	11	.251	4E−3
12	−4.893	4E−3	12	−4.93	6E−2
13	−.19	9E−2	13	−.01	0.11
14	6.96	0.29	14	5.9	1.62
left type (.40, .45 ‖ .42, .47)			$M = 23$		worse than average
right type (.45, .50 ‖ .47, .52)			$M = 18$		semifinal R

The c_n "hump" occurs at about 15 and 14, respectively. The agreement between the c_n's is consistent with our rule-of-thumb. Though the differences are below average in quality, things don't look so bad if one deals with *significant* figures instead. [On a floating-point machine, this might not be such a bad idea ...]

Other cases having relatively large c_n are:

$$R = 25.081315 \quad (\text{e.g. } c_2 = -5.703319, \quad c_3 = -9.051076);$$
$$R = 30.029497 \quad (\text{e.g. } c_2 = -12.67644, \quad c_3 = 22.34885).$$

In order to properly calibrate the output from our *even* runs with $N = 5$ and 7, we made a number of odd runs [using the same N and $\alpha||\beta$] in selected R-ranges beyond 25. The following is an illustration.

Example 14. $R = 50.488237$ ($N = 5$/odd).

$R = 50.48823748$			$R = 50.48823704$		
k	$\frac{1}{2}(c_k' + c_k'')$	rough diff.	k	$\frac{1}{2}(c_k' + c_k'')$	rough diff.
2	−3.814250	8E−6	2	−3.81433	1E−4
3	3.965753	8E−6	3	3.96583	1E−4
4	1.162531	2E−6	4	1.16255	3E−5
5	1.821024	5E−6	5	1.82104	2E−5
6	−1.664537	1E−6	6	−1.66457	7E−5
7	1.012407	2E−6	7	1.01242	3E−5
8	.608971	6E−6	8	.608989	6E−7
9	−.157489	1E−6	9	−.157488	6E−6
10	−1.592329	6E−7	10	−1.592300	5E−6
12	−3.305690	5E−6	12	−3.30577	1E−4
14	2.548589	2E−6	14	2.54862	1E−4
16	3.509777	7E−6	16	3.50983	1E−4
18	2.861370	3E−5	18	2.86141	6E−5
20	−.036687	2E−6	20	−.03668	3E−5
25	4.55179	5E−5	25	4.55174	4E−4
30	−.41234	2E−5	30	−.41239	1E−4
35	3.9363	2E−4	35	3.93652	1E−5
40	.3673	5E−3	40	.3671	2E−3
45	−.18	1.11	45	1.14	0.61
type (.40, .45 || .42, .47)			left $M = 48$ right $M = 50$		a fairly typical case illustrating variation wrt M

The purpose of this example is partly to show how much variation in quality can take place simply by varying M. Both M's are admissible in the sense of §5 and [11, eq.(2.6)]. The *left*-hand column represents the "final" R and is about average in quality.

[The agreement between the c_n's is consistent with our basic rule-of-thumb.]

To further illustrate $N = 5$, it may be useful to take a look at some typical "output" for the case of *even R*. We do so in the following example. The *contrast* between odd and even speaks for itself. [Cf. §6 items (a)–(d).]

Example 15. $R = 48.244$ ($N = 5$/even).

$R = 48.244655$			$R = 48.244535$		
k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	−2.0084	1E−3	2	−2.0389	5E−3
3	.4828	6E−5	3	.5085	3E−6
4	−2.0096	2E−3	4	−2.0615	7E−3
5	−.5300	1E−3	5	−.5457	5E−3
6	.2072	2E−5	6	.2035	9E−6
7	−1.4332	2E−3	7	−1.4718	9E−3
8	.7030	2E−3	8	.735	1E−2
9	.4987	9E−5	9	.5158	3E−4
10	2.2295	3E−3	10	2.264	1E−2
20	.994	1E−1	20	1.028	5E−2
25	.016	1E−2	25	.038	8E−2
30	.093	9E−2	30	.030	2E−1
type $(.40, .45 \,\|\, .42, .47)$ $M = 47$			type $(.40, .45 \,\|\, .42, .47)$ $M = 48$		

$R = 48.244524$			$R = 48.247737(!)$		
k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	−2.037	2E−2	2	−2.075	2E−2
3	.492	2E−2	3	.590	6E−6
4	−2.037	2E−2	4	−2.197	4E−2
5	−.537	4E−2	5	−.602	8E−3
6	.208	4E−3	6	.186	3E−4
7	−1.462	6E−2	7	−1.593	2E−2
8	.717	9E−2	8	.869	1E−2
9	.504	1E−2	9	.592	3E−2
10	2.258	5E−2	10	2.290	7E−3
20	.941	3E−2	20	.969	1E−1
25	−.034	8E−2	25	.051	4E−2
30	.000	4E−2	30	.047	2E−2
type $(.35, .40 \,\|\, .37, .42)$ $M = 57$			type $(.35, .40 \,\|\, .37, .42)$ $M = 58$		

Type $(.40, .45 \| .42, .47)$ *also* included $M = 49$; nothing even remotely resembling 48.244 was picked up there. Similarly for type $(.35, .40 \| .37, .42)$ and $M = 56$.

Output of this kind certainly does *not* give one any reason to hope that $R = 48.244^+$ is an (even) eigenvalue. Things are simply too unstable/fuzzy.

Note that the quality is definitely better for type $(.40, .45 \| .42, .47)$ than for $(.35, .40 \| .37, .42)$. This *agrees* with our earlier comment in §6 about pseudo-residuals.

Examples of this kind were a real nuisance in our production runs. They appeared much more frequently than we originally hoped — and succeeded only in wasting a great deal of CPU time (since the machine was obligated to pursue each one of them down to the level of $H3$).

This completes our tour of interesting "specimens."

§8. Concluding remarks

It remains to wrap up a few loose ends before we close.

(A) In [11], we saw that, when solving (2.5′), it was not generally safe to regard c_n as a smooth function of R at the level of *H3*. This was especially true for larger values of M and R. This state-of-affairs basically stems from a mixture of finite-precision and conditioning effects [on the machine].

In the remarks that follow, let v_k denote the (coarse-grained) rate-of-change of c_k with respect to an R-interval of length *H2*. [It is helpful to think of v_k as a velocity. Though, for the sake of precision, we are referring to (2.5′), the other system can be considered here just as well.]

In situations where v_k does not (yet!) change very rapidly, it is tempting to employ the approximate relation $\Delta\, c_k \cong v_k \cdot \Delta\, R$ in an attempt to draw *additional* accuracy from the *existing* outout [at no extra cost] by making one final interpolation beyond *H*3. Fortunately, in designing our code, we decided[20] to display *both* v_k and its local fluctuation (wrt neighboring *H2* intervals). A quick review of the output files shows that in numerous cases

$$\left|\frac{\text{velocity fluctuation}}{v_k}\right| \ll 1 \qquad \text{for} \qquad 2 \leqq k \leqq M.$$

⟦Small ratios of this type tend to be indicative of good conditioning.⟧

Take Example 1, for instance. Here the foregoing ratio is never bigger than *.003*.[21] The coefficients c_k' and c_k'' can (therefore) be viewed as *linear* functions of R with slope v_k' and v_k''. Let R_0 be the "final" R-value obtained ala §2. We now write

$$R = R_0 + h$$

and consider the equations

$$\{c_2'(R) = c_2''(R), c_3'(R) = c_3''(R), c_4'(R) = c_4''(R), c_5'(R) = c_5''(R), \cdots\}.^{22}$$

The corresponding h-values turn out to be :

---

[20] for reasons of safety

[21] Cf. the excerpt printed below.

[19] as though the contribution form $\{n > M\}$ in (2.4) were exactly 0 !!

$$\{h_2 = 3.765E{-}8, h_3 = 3.561E{-}8, h_4 = 3.599E{-}8, h_5 = 3.588E{-}8, \cdots\}.$$

Note that these h-values are all roughly in the same neighborhood.

Upon taking $h \equiv 3.6E - 8$, we get:

$R = 7.220871975$

k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.	k	$\frac{1}{2}(c'_k + c''_k)$	rough diff.
2	−.707106778	6E−10	7	−.061440	5E−6
3	−.949350733	2E−10	8	−.35283	6E−5
4	.500000082	1E−10	9	−.1044	4E−4
5	−.86971384	2E−8	10	.652	2E−3
6	.6713004	4E−7	11	−.252	7E−3

To gauge the overall accuracy, note that:

$$|c_2 + \frac{1}{\sqrt{2}}| = .000000003 \qquad |c_4 - \frac{1}{2}| = .000000082$$

$$|c_6 - c_2 c_3| = .0000081 \qquad |c_8 + \frac{1}{2\sqrt{2}}| = .00072$$

$$|c_9 - (c_3^2 - 1)| = .0057 \qquad |c_{10} - c_2 c_5| = .037\ .$$

It is clear that we have obtained a substantial increase in accuracy (over Example 1).

Similar refinements can be made in many other cases.

⟦The *essential requirement* is that the velocity fluctuations be small compared to $|v_k|$. Since the K-Bessel functions are only accurate to between 10 and 12 places, 9-place accuracy in c_k is nearing the limit of what we can feasibly hope for.⟧

FOR $R = 7.2208719388$ $\quad |c'_k - c''_k|$ value at k

DIFFS:	0.00000002 (2)	0.00000003 (3)	0.00000045 (4)	0.00000550 (5)
DIFFS:	0.00005861 (6)	0.00054128 (7)	0.00418757 (8)	0.02564663 (9)
DIFFS:	0.11171821 (10)	0.00000000 (12)	0.00000000 (14)	0.00000000 (16)

C(2)=	−0.707106664330198E+00	0.2966E−02	−0.3140E+01	0.9445E−03
C(3)=	−0.949351004190930E+00	0.8271E−02	0.7540E+01	0.1097E−02
C(4)=	0.500001888356337E+00	0.9235E−01	−0.5018E+02	0.1840E−02
C(5)=	−0.869727528347259E+00	0.8678E+00	0.3800E+03	0.2283E−02
C(6)=	0.671406280808934E+00	0.7460E+01	−0.2935E+04	0.2541E−02
C(7)=	−0.622010799211770E−01	0.5714E+02	0.2107E+05	0.2711E−02
C(8)=	−0.348053589509238E+00	0.3735E+03	−0.1319E+06	0.2831E−02
C(9)=	−0.128818742385188E+00	0.1960E+04	0.6708E+06	0.2922E−02
C(10)=	0.742274809428828E+00	0.7394E+04	−0.2472E+07	0.2991E−02
C(11)=	−0.435335184866034E+00	0.1524E+05	0.5001E+07	0.3047E−02
C(2)=	−0.707106679686245E+00	0.2691E−02	−0.2731E+01	0.9852E−03
C(3)=	−0.949351032342392E+00	0.6287E−02	0.8329E+01	0.7548E−03
C(4)=	0.500002342953834E+00	0.9063E−01	−0.6281E+02	0.1443E−03
C(5)=	−0.869733032616470E+00	0.9764E−00	0.5334E+03	0.1831E−02
C(6)=	0.671464893062694E+00	0.9411E+01	−0.4574E+04	0.2057E−02
C(7)=	−0.627423593924994E−01	0.8010E+02	0.3626E+05	0.2209E−02
C(8)=	−0.343866023884553E+00	0.5791E+03	−0.2499E+06	0.2318E−02
C(9)=	−0.154465374877827E+00	0.3351E+04	0.1396E+07	0.2400E−02
C(10)=	0.853993022650567E+00	0.1391E+05	−0.5645E+07	0.2464E−03
C(11)=	−0.698895232302622E+00	0.3151E+08	0.1252E+08	0.2516E−02
		↑	↑	↑
		vel. fluct. (wrt $H2$)	v_k	|ratio|

Part of the original CRAY-2 output for Example 1

(B) As far as the numerics go, the foregoing results all seem very satisfactory. Some readers may wonder, however, how much computer time was actually required. To answer this question, we provide the following sample table.

In interpreting these figures, bear in mind that each job consists of 2 parts: (a) the (unavoidable) portion dealing with level $H2$; (b) the portion stemming from any "blowups" that need to be made at level $H3$.

A quick look at column *6* shows that the relative contribution of (a) and (b) can very *quite* a bit from one job to another. [This fact needs to be taken into account when assigning time limits for the various jobs ...]

	job category	α_1	machine	CPU time	total no.of $H3$ blowups	
1	$N=4$ $R \in [5-l, 10+l]$ odd $M = 10, 11, 12$	.60	CRAY2 (cft)	228 sec.	$2+2+2=6$	
2a	$N=4$ $R \in [20-l, 25+l]$ odd $M = 17, 18, 19$	.60	CRAY2 (cft) XMP (cft77) YMP (cft77)	764 sec. 764 sec. 532 sec.	$11+11+12=34$	ON CRAY-2 time spent
2b	$N=4$ $R \in [20-l, 25+l]$ even $M = 18, 19, 20$	.50	CRAY2 (cft)	903 sec.	$17+13+12=42$	in part (a) $\sim$ 175 sec.
3	$N=5$ $R \in [10-l, 15+l]$ even $M = 23, 24, 25$	.40	CRAY2 (cft)	265 sec.	$0+0+0=0$	
4a	$N=5$ $R \in [20-l, 25+l]$ odd $M = 28, 29, 30$	.40	CRAY2 (cft) XMP (cft77)	1494 sec. 1450 sec.	$14+14+13=41$	ON CRAY-2 time spent
4b	$N=5$ $R \in [20-l, 25+l]$ even $M = 28, 29, 30$	.40	CRAY2 (cft)	1097 sec.	$11+6+8=25$	in part (a) $\sim$ 475 sec.
5	$N=5$ $R \in [45-l, 50+l]$ even $M = 47, 48, 49$	.40	CRAY2 (cft)	2100 sec.	$2+2+2=6$	
6	$N=5$ $R \in [55-l, 60+l]$ even $M = 56, 57, 58$	.40	CRAY2 (cft)	2008 sec.	$0+0+0=0$	
7	$N=6$ $R \in [20-l, 25+l]$ even $M = 31, 32, 33$	.40	CRAY2 (cft)	2714 sec.	$19+26+24=69$	
8	$N=7$ $R \in [30-l, 35+l]$ even $M = 56, 57, 58$	.30	YMP (cft77)	2268 sec.	$12+7+3=22$	
9	$N=7$ $R \in [35-l, 40+l]$ even $M = 62, 63, 64$	.30	YMP (cft77)	10594 sec.	$41+41+33=115$	
10	$N=5$ $49.875 \le R \le 51.000$ odd $M = 48, 50$ test run	.40	CRAY2 (cft) CRAY2 (cft77) XMP (cft77) YMP (cft77)	1366 sec. 1270 sec. 1370 sec. 966 sec.	$7+7=14$	
11	$N=7$ $50.125 \le R \le 51.25$ odd $M = 75, 76, 77$ test run	.30	CRAY2 (cft)	5882 sec.	8+8+10=26	
	type $(\alpha_1, \alpha_2 \,\|\, \beta_1, \beta_2)$		$H2 = .001$, $H3 = 10^{-6}$		$l \equiv \frac{1}{8}$	

On the basis of this table, the following speed ratios are seen to apply:

$$\frac{\text{YMP(cft77)}}{\text{CRAY2(cft)}} \cong 1.42, \quad \frac{\text{YMP(cft77)}}{\text{CRAY2(cft77)}} \cong 1.31, \quad \frac{\text{YMP(cft77)}}{\text{XMP(cft77)}} \cong 1.42, \quad ^{26}$$

In all:

$$\left\{\begin{matrix} 110 \\ 14 \\ 5 \end{matrix}\right\} \text{ of our type } (\alpha_1, \alpha_2 \| \beta_1, \beta_2) \text{ jobs used the } \left\{\begin{matrix} \text{CRAY2} \\ \text{XMP} \\ \text{YMP} \end{matrix}\right\}.$$

[20] As usual, these figures represent a *composite* of both the memory and clock speeds.

The corresponding CPU totals were:

	odd $N=4$	odd $N=5$	odd $N=6$	odd $N=7$	even $N=4$	even $N=5$	even $N=6$	even $N=7$	
CRAY2	3962	10840	4820	8319	9351	35337	22313	12747	107689 sec.
XMP	764	5553	8172	0	0	0	0	0	14489 sec.
YMP	532	966	0	0	0	0	0	16231	17729 sec.

(C) We finish up by drawing attention to several possibilities for further work.

(α) *Linear Algebra.* Our subroutine for solving linear equations is nothing but standard Gauss elimination. It can certainly be improved. For values of M bigger than 60 or so, it would probably be best to switch over to one of the optimized routines available in a standard library. [This would help cut the CPU time!] Creative use of iterative techniques is another possibility.

(β) *Additional Coefficients.* For arithmetic $\mathbf{G}_N$, it would be quite useful to obtain many more Fourier coefficients than we currently have. This can probably be done by implementing some version of H. Stark's method [35].[30]

In *non*arithmetic cases, this problem is also quite interesting. Here, however, it is not so clear what to do. Further analysis seems very much in order.

(γ) *Other Groups.* It goes without saying that one would very much like to run similar R_n-experiments for more general Γ (and for more general multiplier systems). ⟦Though in this paper we have focussed only on $\mathbf{G}_N$, the numerical groundwork we've laid can clearly be extended to much more general groups. Any subtleties that occur for $\mathbf{G}_N$ can be expected to *recur* (possibly with a vengeance!) when more general Γ are used. For this reason, the case of $\mathbf{G}_N$ is actually an important testing-ground.⟧

(δ) *Completeness Questions.* In the absence of any kind of (numerical) argument principle for $Z_\Gamma(s)$, it is a bit irritating that one cannot rigorously say when one is done. Can this deficiency be corrected ?? Cf. the suggestions at the end of §3.

[30]For an update on progress in this area, see [15].

References

[1] A.O.L. Atkin and J. Lehner, Hecke operators on $\Gamma_0(m)$, Math. Ann., **185** (1970), 134–160.

[2] N.L. Balazs and A. Voros, Chaos on the pseudosphere, Physics Rep., **143** (1986), 109–240.

[3] J.M. Deshouillers, H. Iwaniec, R.S. Phillips and P. Sarnak, Maass cusp forms, Proc. National Acad. Sci. U.S.A., **82** (1985), 3533–3534.

[4] A. Erdélyi, et. al., "Higher Transcendental Functions", vol.2, McGraw-Hill, 1953, pp. 86–88, esp. 86(7), 87(18), 88(19)(20).

[5] L. Fox, P. Henrici, C. Moler, Approximations and bounds for eigenvalues of elliptic operators, SIAM J. Num. Anal., **4** (1967), 89–102.

[6] G.H. Golub and C.F. Van Loan, "Matrix Computations", Johns Hopkins Univ. Press, 1983.

[7] L. Greenberg, Maximal groups and signatures, in *Discontinuous Groups and Riemann Surfaces*, Annals of Math. Studies, No. **79** (1974), 207–226. See also: Bull. Amer. Math. Soc. **69** (1963), 569–573

[8] E. Hecke, "Mathematische Werke", Vandenhoeck & Ruprecht, Göttingen, 1959.

[9] D.A. Hejhal, "The Selberg Trace Formula for PSL(2, **R**)", vol.1, Springer Lecture Notes No. 548, 1976.

[10] ———, "The Selberg Trace Formula for PSL(2, **R**)", vol.2, Springer Lecture Notes, No. 1001, 1983.

[11] ———, Eigenvalues of the Laplacian for PSL(2, **Z**): some new results and computational techniques, in *International Symposium in Memory of Hua Loo-Keng* (ed. by S. Gong, Q. Lu, Y. Wang, L. Yang), Science Press and Springer-Verlag, 1991, vol.1, pp.59–102.

[12] ———, Eigenvalues of the Laplacian for Hecke triangle groups, Amer. Math. Soc. Memoirs, to appear, 1992.

[13] ———, Some observations concerning eigenvalues of the Laplacian and Dirichlet L-series, in *Recent Progress in Analytic Number Theory*, vol. 2, Academic Press, 1981, 95–110.

[14] ———, Zeros of Epstein zeta functions and supercomputers, in *Proc. of the International Congr. of Math.*, Berkeley 1986, pp.1362–1384.

[15] D.A. Hejhal and S. Arno, Fourier coefficients for Maass waveforms on PSL(2, **Z**), preprint, 1991.

[16] D.A. Hejhal and B. Berg, Some new results concerning eigenvalues of the non-Euclidean Laplacian for PSL(2, **Z**), Univ. of Minn. Math. Report No. 82-172, 1982, 7pp.

[17] A. Hurwitz, Über eine Reihe neuer Funktionen, welche die absoluten Invarianten gewisser Gruppen ganzzahliger linearer Transformationen bilden, Math. Ann., **20** (1882), 125–134.

[18] H. Iwaniec, Non-holomorphic modular forms and their applications, in *Modular Forms* (ed. by R.A. Rankin), Ellis-Horwood, 1984, pp.157–196, especially 178–180.

[19] J. Lehner, "Discontinuous Groups and Automorphic Functions", AMS Math. Surveys, No. 8, 1964.

[20] A. Leutbecher, Über die Heckeschen Gruppen $\mathbf{G}(\lambda)$, Abh. Math. Sem. Hamburg, **31** (1967), 199-205.

[21] T. Miyake, On automorphic forms on GL(2) and Hecke operators, Ann of Math., **94** (1971), 174–189.

[22] H. Petersson, Über die eindeutige Bestimmung und die Erweiterungsfähigkeit von gewissen Grenzkreisgruppen, Abh. Math. Sem. Hamburg, **12** (1938), 180–199, especially 184 (middle), 193 (Satz 4). Also: E. Schulenberg, same journal, **13** (1939), 144–199.

[23] R.S. Phillips and P. Sarnak, On cusp forms for cofinite subgroups of $PSL(2, \mathbf{R})$, Invent. Math., **80** (1985), 339–364.

[24] ———, The Weyl theorem and the deformation of discrete groups, Comm Pure Appl. Math., **38** (1985), 853–866.

[25] B. Randol, A remark on the multiplicity of the discrete spectrum of congruence groups, Proc. Amer. Math. Soc., **81** (1981), 339–340.

[26] R.A. Rankin, "Modular Forms and Functions", Cambridge Univ. Press, 1977, pp. 245–342.

[27] O. Rausenberger, Zur Theorie der Funktionen mit mehreren, nicht vertauschbaren Perioden, Math. Ann., **20** (1882), 47–48.

[28] P. Sarnak, On cusp forms, Contemp. Math., **53** (1986), 393–407.

[29] ———, Determinants of Laplacians, Comm. Math. Phys., **110** (1987), 113–120.

[30] A. Selberg, Harmonic analysis and discontinuous groups in weakly symmetric Riemannian spaces with applications to Dirichlet series, J. Ind. Math. Soc., **20** (1956), 47–87.

[31] ———, Discontinuous groups and harmonic analysis, in *Proc. of the Internat. Congr. of Math.*, Stockholm 1962, pp.177–189.

[32] ———, Remarks on the distribution of poles of Eisenstein series, to appear in "Collected Papers," vol.2, Springer-Verlag, 1992.

[33] G. Shimura, "Introduction to the Arithmetic Theory of Automorphic Functions", Princeton Univ. Press, 1971, especially §3.4, 3.5.

[34] D. Singerman, Finitely maximal Fuchsian groups, J. London Math. Soc., **6** (1972), 29–38.

[35] H. Stark, Fourier coefficients of Maass waveforms, in *Modular Forms* (ed. by R.A. Rankin), Ellis-Horwood, 1984, pp.263–269.

[36] K. Takeuchi, Arithmetic triangle groups, J. Math. Soc. Japan, **29** (1977), 91–106. Also: J. Fac. Sci. Univ. Tokyo **24** (1977), 201–212.

[37] A.B. Venkov, Spectral theory of automorphic functions, Proc. Steklov Inst. Math., **153** (1982).

[38] A. Voros, Spectral functions, special functions, and the Selberg zeta function, Comm. Math. Phys., **110** (1987), 439–465.

[39] A. Winkler, Cusp forms and Hecke groups, J. Reine Angew. Math., **386** (1988), 187–204.

[40] ———, Dirichlet and Neumann spectrum, Comm. Pure Appl. Math., **41** (1988), 305–316.

School of Mathematics
University of Minnesota
Minneapolis, MN 55455
U.S.A.