Computing and Verifying Maass Forms

David Lowry-Duda

February 2021

ICERM
Brown University

Acknowledgements

This is a project I've begun since joining the Simons Collaboration on Arithmetic Geometry, Number Theory, and Computation. I've collected a large amount of data associated to Maass forms, but there remains a lot to compute and a lot to prove.

In this talk, I'll touch on work done with several collaborators. In particular, I've been working with Andrew Booker (Bristol) and Drew Sutherland (MIT) on computational aspects, and Min Lee, Jonathan Bober, Andrei Seymour-Howell, and Andrew Booker (all at Bristol) with theoretical aspects.

I should also note that I've had the benefit of several helpful conversations with David Farmer (AIM), Sally Koutsoliotas (Bucknell), Stefan Lemurell(Chalmers), Fredrik Strömberg (Nottingham), and the rest of the Simons Collaboration.

Talk overview

Maass forms

Hejhal's algorithm

Linearization, Turing's method and large scale computation

Rigorous verification

Maass forms

Maass forms

Hejhal's algorithm

Linearization, Turing's method and large scale computation

Rigorous verification

Motivation: why study Maass forms?

Maass forms are solutions to the real-analytic eigenvalue problem of the Laplacian on modular surfaces. They're both highly structured and very mysterious.

Maass forms extend the classical theory of Dirichlet series with Euler products and the theory of classical holomorphic modular forms. The spectral theoretic decomposition into Maass forms led to the discovery of Selberg's trace formula, which connects the spectrum to the underlying geometry.

Personally, I frequently use spectral theory and poor understanding of Maass forms is the most common major obstruction I face.

For this talk, a Maass form will be a *weight* 0 *Maass cuspform* on a congruence subgroup of $\mathrm{SL}(2,\mathbb{Z})$. Specifically, let $\Gamma < \mathrm{SL}(2,\mathbb{Z})$ be a congruence subgroup. The modular surface $X = \Gamma\backslash\mathcal{H}$ is a finite non-compact surface. The Laplacian Δ on this surface is $\Delta = -y^2(\partial^2/\partial x^2 + \partial^2/\partial y^2)$.

We call a function $f : \mathcal{H} \longrightarrow \mathbb{C}$ a Maass cuspform if

1. f is real analytic, $f \in C^\infty(\mathcal{H})$,
2. f is an eigenfunction of the Laplacian, $\Delta f = \lambda f$,
3. f is automorphic, $f(\gamma z) = f(z)$ for all $\gamma \in \Gamma$,
4. f is square integrable, $f \in L^2(X)$, and
5. f vanishes at all the cusps of X.

Selberg famously conjectured that (for congruence subgroups Γ) that the eigenvalue λ is either 0 or $\lambda \geq \frac{1}{4}$. An eigenvalue $\lambda \in (0, \frac{1}{4})$ would be called *exceptional*, though we've never seen one.

This *Selberg eigenvalue conjecture* (SEC) is analogous to the Ramanujan–Petersson Conjecture (RPC). We describe this now.

Given a classical (weight k Hecke) holomorphic cusp form

$$g(z) = \sum_{n\geq 1} a(n) n^{\frac{k-1}{2}} e^{2\pi i n z},$$

one can associate an L-function

$$L(s,g) = \sum_{n\geq 1} \frac{a(n)}{n^s} = \prod_p L_p(s),$$

where $L_p(s)$ is (generically) of the form

$$L_p(s) = (1 - \beta_{p,1} p^{-s})^{-1} (1 - \beta_{p,2} p^{-s})^{-1}.$$

The RPC asserts that $|\beta_{p,j}| = 1$, or equivalently that $\log_p|\beta_{p,j}| = 0$.

For holomorphic cusp forms, the RPC is known and follows from Deligne's celebrated proof [Del71].

To each Maass form, there is also an associated L-function. In its completed form, the L-function associated to a Maass form f has the shape

$$\Lambda(s, f) = L_\infty(s) \prod_p L_p(s),$$

where (for generic p)

$$\begin{aligned} L_p(s) &= (1 - \alpha_{p,1} p^{-s})^{-1} (1 - \alpha_{p,2} p^{-s})^{-1} \\ L_\infty(s) &= \Gamma_\mathbb{R}(s - \mu_{\infty,1}) \Gamma_\mathbb{R}(s - \mu_{\infty,2}). \end{aligned}$$

Here, $L_\infty(s)$ is the "factor at ∞" and consists of a pair of gamma functions $\Gamma_\mathbb{R}(s) := \pi^{-s/2}\Gamma(s/2)$.

The parameters $\mu_{\infty,j}$ are closely related to the eigenvalues, and SEC states that $\operatorname{Re} \mu_{\infty,j} = 0$ while RPC states that $\log_p |\alpha_{p,j}| = 0$.

The best progress towards these conjectures for Maass forms are due to Kim and Sarnak, who showed that $|\operatorname{Re} \alpha_{\infty,j}|$ and $\big| \log_p |\alpha_{p,j}| \big|$ are bounded above by $\frac{7}{64}$ [KS03].

Finally, each function $g \in L^2(\Gamma\backslash\mathcal{H})$ has a spectral expansion of the shape

$$\begin{aligned} g(z) = &\sum_{f \text{ Maass cuspform}} \langle g, f\rangle f(z) \\ &+ \sum_{\text{Eisenstein}} \int \langle g, E(\cdot, u)\rangle E(z, u) du \\ &+ \text{(a constant)}. \end{aligned}$$

My most common *hammer* in my *tool belt* is to take averages, represent everything in terms of the spectral decomposition, and roll up my sleeves and do complex analysis on what remains. The Maass forms that appear in these expansions are typically the barrier to better results.

(This has been the case in all but one of my papers with my frequent collaborator Alex Walker).

Maass forms in the LMFDB

The L-function and modular form database (`https://LMFDB.org`) is an online database of L-functions, modular forms, abelian varieties, and their relationships.

There is currently heuristic data for nearly 15000 Maass forms in the LMFDB, available through the portal `https://www.lmfdb.org/ModularForm/GL2/Q/Maass/`. But we know how to compute more data and to make these computations rigorous.

Hejhal's algorithm

Maass forms

Hejhal's algorithm

Linearization, Turing's method and large scale computation

Rigorous verification

Methods for the computation of Maass forms have been considered and developed by several authors since the 1970s. Today, I'll describe my preferred method (for GL(2) type Maass forms): Hejhal's algorithm.

In my experience, Hejhal's algorithm is faster and more accurate compared to earlier methods. On the other hand, Hejhal's algorithm is *not rigorous* (although in practice it always produces reliable results). We'll return to the topic of rigorous evaluation later.

The algorithm that Hejhal described apply for the computation of Maass forms for cofinite Fuchsian groups Γ such that $\Gamma\backslash\mathcal{H}$ has exactly one cusp, but I'll also describe the necessary adjustments for when $\Gamma\backslash\mathcal{H}$ has multiple cusps, as is the case for general congruence subgroups Γ.

Maass form Fourier expansion

It is easiest to first describe using Hejhal's algorithm to compute a known Maass newform. Let us fix a Maass form f with eigenvalue $\lambda = \frac{1}{4} + R^2$. Then f has a Fourier expansion

$$f(z) = \sum_{n \geq 0} c(n) \sqrt{y} \kappa_{iR}(2\pi |n| y) e(nx). \tag{1}$$

Here and later, we use the notation $e(nx) = e^{2\pi i n x}$ and $\kappa_{iR}(u) = e^{\pi R/2} K_{iR}(u)$, where $K_\alpha(u)$ is the modified K-Bessel function of the second kind.

In this normalization, $\kappa_{iR}(u)$ is an oscillating function of u for $0 < u \lesssim R$ with amplitude roughly of size 1, and then it decays exponentially for $u \gtrsim R$.

Note that in terms of (1), we interpret our goal of *computing a Maass form* to mean to *find the eigenvalue parameter R and the coefficients $c(n)$.*

The coefficients $c(n)$ satisfy the trivial Hecke bound $c(n) = O(\sqrt{n})$ (in fact, much better bounds are known). We can further assume that $c(1) = 1$. Let us fix a desired error bound 10^{-D}. Then there is a decreasing function $M(y) = M(y, R)$ such that

$$f(x + iy) = \sum_{|n| \leq M(y)} c(n)\sqrt{y}\kappa_{iR}(2\pi|n|y)e(nx) + [[10^{-D}]],$$

(where we use $[[10^{-D}]]$ to mean a quantity of absolute value strictly less than 10^{-D}).

Thus we can view $f(x + iy)$ as a finite Fourier series in x up to a small, controlled error.

$$f(x+iy) = \sum_{|n| \leq M(y)} c(n)\sqrt{y}\kappa_{iR}(2\pi|n|y)e(nx) + [[10^{-D}]].$$

The finite Fourier series part of the sum is essentially a discrete Fourier transform. If we choose equally spaced points along a horocycle

$$\{z_m = x_m + iY : x_m = \frac{1}{2Q}(m - \tfrac{1}{2}), 1 - Q \leq m \leq Q\},$$

(with $Q > M(Y)$), then we can invert this transform to see that

$$c(n)\sqrt{Y}\kappa_{iR}(2\pi|n|Y) = \frac{1}{2Q}\sum_{1-Q=m}^{Q} f(z_m)e(-nx_m) + [[10^{-D}]].$$

For fixed R and Y, we can vary n to get essentially a linear system in the coefficients $c(n)$ — but this system is currently a tautology.

We make this system non-tautological by using the automorphy of f, that $f(\gamma z) = z$ for all $\gamma \in \Gamma$. To accomplish this, for the points $z_m = x_m + iY$ in our horocycle, we choose Y small enough so that part of the horocycle will be outside fixed fundamental domain for $\Gamma\backslash\mathcal{H}$.

Then we pullback each z_m to a point z_m^* in the fundamental domain. The result is that

$$c(n)\sqrt{Y}\kappa_{iR}(2\pi|n|Y) = \frac{1}{2Q}\sum_{1-Q=m}^{Q} f(z_m)e(-nx_m) + [[10^{-D}]].$$

becomes

$$c(n)\sqrt{Y}\kappa_{iR}(2\pi|n|Y) = \frac{1}{2Q}\sum_{1-Q=m}^{Q} f(z_m^*)e(-nx_m) + [[10^{-D}]].$$

If instead of a congruence subgroup, we were considering $\mathrm{SL}(2,\mathbb{Z})\backslash\mathcal{H}$, we would be done. We could expand each $f(z_m^*)$ in its own (essentially finite) Fourier series, repeat for several n, and get a linear system with unknowns $c(n)$. This is the classical algorithm of Hejhal.

Expansions at all the cusps

But when $\Gamma\backslash\mathcal{H}$ has multiple cusps, the resulting linear system is typically very poorly-conditioned. Heuristically this is because several points $z_m = x_m + iY$ might still be in the fundamental domain, and thus $f(z_m) = f(z_m^*)$ for these points — the system is insufficiently mixed by the modularity.

To resolve this, we work not just with the Fourier expansion of f at ∞. We instead work simultaneously with the Fourier expansions f_ℓ at each cusp ℓ. That is, in terms of the Fourier expansions $f_\ell(z) = f(\sigma_\ell z)$, where $\sigma_\ell \infty = \ell$ is a cusp normalization map.

For each point z^* in the fundamental domain, we identify the nearest cusp $\ell = \ell(z^*)$. (By nearest, we mean the cusp with respect to which z^* has the greatest height). Then we represent the value $f(z^*)$ in terms of the Fourier expansion f_ℓ.

(This is the lots-of-bookkeeping aspect of the approach). In order to set up the extended system, we must enlarge our linear system to include horocycles associated to the expansion at each cusp and solve for all expansions simultaneously. For each cusp j, we have an expansion

$$f_j(z) = \sum_{n \neq 0} c_j(n) \sqrt{y} \kappa_{iR}(2\pi |n| y) e(nx)$$

and we can set up the system

$$c_j(n) \sqrt{Y} \kappa_{iR}(2\pi |n| Y) = \frac{1}{2Q} \sum_{1-Q=m}^{Q} f_j(z_m) e(-nx_m) + [[10^{-D}]]$$

as before.

We now have the system

$$c_j(n)\sqrt{Y}\kappa_{iR}(2\pi|n|Y) = \frac{1}{2Q}\sum_{1-Q=m}^{Q} f_j(z_m)e(-nx_m) + [[10^{-D}]].$$

Let $z_{mj} = \sigma_j z_m$, so that $f_j(z_m) = f(z_{mj})$, and let z_{mj}^* be the pullback of z_{mj} to the fundamental domain, expressed in coordinates of the nearest cusp ℓ. Then we recognize $f(z_{mj})$ as $f_\ell(z_{mj}^*)$, and in total

$$c_j(n)\sqrt{Y}\kappa_{iR}(2\pi|n|Y) = \frac{1}{2Q}\sum_{1-Q=m}^{Q} f_\ell(z_{mj}^*)e(-nx_m) + [[10^{-D}]].$$

Lemma

It is possible to choose Y small enough such that $z^*_{mj} \neq z_{mj}$ for all j and m. Further, the imaginary parts of each resulting z^*_{mj} are bounded below by a computable constant Y_0 (which depends on the level of the congruence subgroup).

It is the nontrivial mixing coming from $f_j(z_m)$ and $f_\ell(z^*_{mj})$ that gives a non-tautological system, allowing us to solve for the Fourier coefficients in the linear system.

Solving for the coefficients

In summary, given an input eigenvalue $\lambda = \frac{1}{4} + R^2$, we can set up the system

$$c_j(n)\sqrt{Y}\kappa_{iR}(2\pi|n|Y) = \frac{1}{2Q}\sum_{1-Q=m}^{Q} f_\ell(z_{mj}^*)e(-nx_m) + [[10^{-D}]].$$

If we choose the Y in the horocycles as in the Lemma, then $\mathrm{Im}(z_{mj}^*) > Y_0$ for all m and j, so we can truncate each Fourier series f_ℓ on the right at the same point $M_0 = M(Y_0)$ while guaranteeing a uniform error bound. Expanding each finite Fourier series and collecting coefficients, we get that

$$c_j(n)\sqrt{Y}\kappa_{iR}(2\pi|n|Y) = \sum_{\text{cusps}\,\ell}\ \sum_{1\le|k|\le M_0} c_j(k)V_{nkj\ell} + 2[[10^{-D}]]$$

for complicated-but-computable coefficients $V_{nkj\ell}$ (that are just complicated combinations of K-Bessel functions). Considering this for each $|n| \le M_0$ gives a linear system that can be solved.

Structurally, we have constructed a homogeneous linear system $V\vec{c} = 0$ for a computable matrix $V = V(R, Y)$ consisting mostly of linear combinations of Bessel functions and an unknown vector of coefficients $\vec{c}$.

We can use the assumption $c(1) = 1$ to de-homogenize the linear system and to facilitate solving for the coefficients.

It should be noted that a priori, it is not obvious that the resulting linear system will be well-conditioned. This would be a necessary ingredient to conclude that this algorithm would always succeed, but this is unknown. However, in my experiments it seems that whenever we choose Y small enough so that $z_{mj} \neq z^*_{mj}$ for all m and j, the resulting system is solvable and gives approximately D correct digits of accuracy for the coefficients.

There are frequently relations between the cusps that allow one to reduce the dimension of the linear system. In particular, there are Hecke-operator type symmetries (Fricke involutions) that connect Fourier expansions at cusps.

I'll also remark that all the work here carries through even when there is a nontrivial nebentypus, except that one must track the character and how it carries through the cusp-normalizing maps σ_ℓ. (This is simply additional bookwork).

Linearization, Turing's method and large scale computation

In the description of Hejhal's algorithm, we initially began with a known eigenvalue $\lambda = \frac{1}{4} + R^2$. It remains to actually find these eigenvalues.

In fact, the algorithm above works for any R and yields a homogeneous linear system $V(R, Y)\vec{c} = 0$. When R comes from a true eigenvalue, the resulting coefficients should be independent of the input Y (as long as Y is small enough for the Lemma to apply). In practice, when R is far from a true eigenvalue, it appears that the resulting coefficients vary rapidly as Y changes.

One approach to find the actual eigenvalues (which was the approach I was using until this January) would be to create a *cost function* $\mathrm{cost}(R)$ that is large when R is far from a true eigenvalue and small when R is (presumably) near a true eigenvalue, evaluate $\mathrm{cost}(R)$ on a mesh, and minimize it. In practice, this worked extremely well if I had a good initial approximation to an eigenvalue, but it was computationally expensive to repeatedly run to try to find initial approximations.

I've instead moved to linearization as a tool to find eigenvalues.

Abstractly, we can rephrase this goal as trying to determine R so that the linear system $V(R, Y)\vec{c} \approx 0$ has nontrivial solutions $\vec{c}$ and for which these solutions are independent of the height Y of the horocycle (which I now suppress from the notation). Given a guess $\widetilde{R}$, we can linearize with respect to R and write

$$V(\widetilde{R} + h)\vec{c} = \Big(V(\widetilde{R}) + hV'(\widetilde{R}) + \frac{h^2}{2}\mathrm{Err}(R, h)\Big)\vec{c}.$$

$$V(\widetilde{R}+h)\vec{c} = \Big(V(\widetilde{R}) + hV'(\widetilde{R}) + \frac{h^2}{2}\mathrm{Err}(R,h)\Big)\vec{c}.$$

If the error weren't there, we could rewrite $V(\widetilde{R}+h)\vec{c} \approx 0$ as

$$V'(\widetilde{R})^{-1}V(\widetilde{R})\vec{c} = -h\vec{c}.$$

If $V'(R)$ is nonsingular, then solving for h becomes a question of determining eigenvalues of the LHS. Solving for the smallest eigenvalue h gives a new approximate eigenvalue $\widetilde{R}+h$. The approximation can be refined iteratively to yield an eigenvalue.

In practice, if $\widetilde{R}$ is close to a true eigenvalue R, then this iterative refinement gives a good estimation of a true eigenvalue.

To find several eigenvalues, one would then choose a mesh of candidates

$$0 < \widetilde{R_1} < \widetilde{R_2} < \widetilde{R_3} < \cdots < \widetilde{R_{max}}$$

sufficiently close together (based on the Weyl law and expected differences between eigenvalues, for instance), linearizing, and iteratively improving.

There are several caveats, but this technique has been employed by Holger Then to compute 200000 eigenvalues of Maass cusp forms on $\mathrm{SL}(2, \mathbb{Z}) \backslash \mathcal{H}$ [The12], and I'm currently adjusting this for higher level forms.

What about missing eigenvalues?

At several steps, we have followed heuristic evaluations. But it is inevitable that we will miss some eigenvalues. In order to detect whether there are eigenvalues missing from a collection of eigenvalues, we can turn to Weyl's Law.

Let

$$N(r) = \#\{\lambda : \tfrac{1}{4} \leq \lambda \leq \tfrac{1}{4} + r^2 \|$$

count the number of eigenvalues in the interval $[\frac{1}{4}, \frac{1}{4} + r^2]$. The Average Weyl's law says that

$$N(r) = M(r) + S(r),$$

where $M(r)$ is a smooth *main term* approximation and the average value of the Weyl remainder $S(r)$ tends to 0 in the limit as $r \to \infty$.

It is possible to derive Weyl's law explicitly. For example, on the classic modular surface:

Theorem (Average Weyl's Law)
On $\mathrm{SL}(2,\mathbb{Z})\backslash\mathcal{H}$*, we have*

$$M(r) = \frac{1}{12}r^2 - \frac{2r}{\pi}\frac{r}{e\sqrt{\pi/2}} - \frac{131}{144}.$$

It is sometimes also possible to derive Turing bounds for the error.

Theorem (Booker and Strömbergson)
Define

$$E(r) = \left(1 + \frac{6.6}{\log r}\right)\left(\frac{\pi}{12\log r}\right)^2.$$

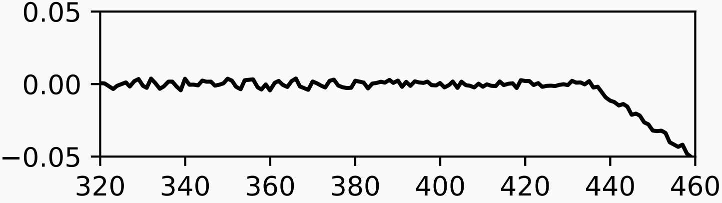
Then $\mathrm{SL}(2,\mathbb{Z})\backslash\mathcal{H}$*, we have that*

$$-2E(r) < \frac{1}{r}\int_0^r S(r)dr < E(r).$$

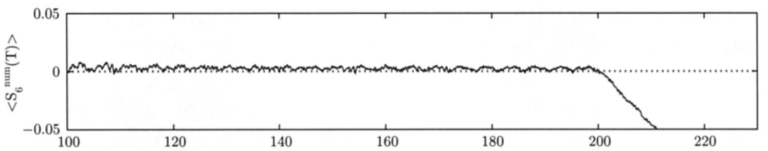
Let's see how this works in practice. The error term $S(r)$ oscillates around zero. Let $N^{found}(r)$ count the number of found eigenvalues in $[1/4, 1/4 + r^2]$. Then we consider

$$S^{found}(r) := N^{found}(r) - M(r) \approx S(r).$$

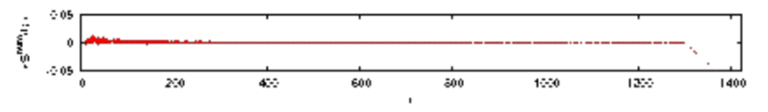
Once an eigenvalue is missed, $S^{found}(r)$ deviates sharply from the otherwise small $S(r)$. On a graph of the mean value of S^{found}, this looks like



This is approach frequently employed by Holger Then in his computations of Maass forms. Detecting a missed eigenvalue on the moonshine group $\Gamma_0(6)^+$:



Or detecting missed eigenvalues for $\mathrm{SL}(2,\mathbb{Z})\backslash\mathcal{H}$:



When Turing bounds are available, these methods can be used to prove that no eigenvalues have been missed. But for most congruence subgroups, there aren't known Turing bounds and deriving them seems difficult.

This is a problem I'm working on with my collaborators, but we haven't fully resolved this rigorously yet.

But heuristically, with an average Weyl law (or even with a heuristic average Weyl law), this method works pretty well.

First 50 eigenvalues for $\mathrm{SL}(2,\mathbb{Z})$

91.1413453	148.432131	190.131547	206.416795	260.687405
277.281364	314.906630	330.795773	377.521632	379.904073
404.529171	454.613156	461.599913	492.853552	519.447625
538.554784	541.450274	581.655554	596.572502	627.795328
667.244864	679.212963	684.181573	699.693619	744.687610
747.326931	771.751770	813.085934	814.253563	833.345533
849.249008	873.239051	994.175385	996.679743	1057.02772
1082.07907	1158.14690	1187.48464	1260.66683	1284.87576
1345.49289	1358.64049	1430.98609	1467.39096	1534.38888
1553.20356	1582.19105	1644.01332	1655.81757	1727.11603

First 50 eigenvalues for $\Gamma_0(5)$

9.42106297	17.086429	17.326764	24.232910	29.802058
32.8064563	36.901259	40.588036	41.961917	46.810520
53.9133074	57.787055	64.034965	64.546059	67.919477
69.0520113	72.160668	78.005312	78.055469	86.318254
88.5614566	96.946006	101.16851	105.23669	106.48480
110.509306	113.27141	120.89757	123.30748	124.66026
125.273302	127.56603	134.82249	136.96036	141.16706
144.689979	147.16315	151.88883	159.92051	160.48779
161.944167	165.51281	169.26726	171.76540	175.01274
181.663468	184.53461	186.50137	187.92101	189.58619

Rigorous verification

Let us now suppose that we have used Hejhal's algorithm (or possibly another algorithm) to determine a possible eigenvalue $\widetilde{\lambda} = \frac{1}{4} + \widetilde{R}^2$ and the coefficients of a possible Maass form $\widetilde{f}$ with that eigenvalue.

In 2006, Booker, Strömbergson, and Venkatesh proved that it is possible to *certify* whether this candidate Maass form is in fact "very close" to a true Maass form. [BSV06]

In particular, over $\mathrm{SL}(2, \mathbb{Z})$ they showed that if $\widetilde{f}$ is "almost automorphic" (i.e. almost invariant under the group action), then $\widetilde{f}$ is "very close" to a true Maass cusp form f.

Small bit of intuition

The idea underlying their proof is that if

$$\|(\Delta - \widetilde{\lambda})\widetilde{f}\|_2^2 \tag{2}$$

is small, then by the spectral expansion almost all of the spectral support of $\widetilde{f}$ is concentrated near $\widetilde{\lambda}$. (This is true for general square integrable functions $\widetilde{f}$ as well).

Then the task is to determine bounds for (2). For a few technical reasons, it ends up being necessary to determine bounds for a smoothed version $\widetilde{f}_S$ (smoothed by convolving with a certain kernel function).

By virtue of the Fourier expansion in K-Bessel functions of $\widetilde{f}$, we get that $(\Delta - \widetilde{\lambda})\widetilde{f}$ vanishes on the fundamental domain, and is invariant under translation by $\mathbb{Z}$ also due to the Fourier expansion. For the smoothed version $\widetilde{f}_S$, this is true except in a small neighborhood of the arc at the bottom of the fundamental domain.

By making these bounds explicit, one can prove the following theorem.

Theorem (BSV)

Let $B(\delta)$ be a hyperbolic δ-neighborhood of the arc $\{z \in \mathcal{H} : |z| = 1, |\mathrm{Re}\, z| \leq 1/2\}$, and let $\widetilde{f}_\Gamma$ denote the $\mathrm{SL}(2, \mathbb{Z})$-periodized extension of $\widetilde{f}$ from the fundamental domain to $\mathcal{H}$.

With the notations as above, there exists a true Maass cusp form on $\mathrm{SL}(2, \mathbb{Z})\backslash\mathcal{H}$ with eigenvalue λ satisfying

$$|\lambda - \widetilde{\lambda}| < C(\widetilde{f}, \delta, \widetilde{R}) \text{ess sup}_{z \in B(\delta)} |\widetilde{f}(z) - \widetilde{f}_\Gamma(z)|$$

for a computable constant $C(\widetilde{f}, \delta, \widetilde{R})$.

And thus to certify a candidate Maass form on the full modular group, it suffices to compute the constant C and bound the deviation from proper automorphicity.

Ongoing work

BSV implemented this to certify the first 10 eigenvalues on $\mathrm{SL}(2, \mathbb{Z})$ to over 1000 decimal places (and analyzed algebraic properties and transcendentality of the numbers).

I'm currently working on large scale (lower quality but faster) verification for Maass forms.

I hope to have completed heuristic computation for many congruence subgroups soon, with additional computation verification shortly afterwards.

Thank you very much.

Please note that these slides (and references for the cited works) are (or will soon be) available on my website (davidlowryduda.com).

References i

Andrew R Booker and David J Platt.
Turing's method for the selberg zeta-function.
Communications in Mathematical Physics, 365(1):295–328, 2019.

Andrew R Booker, Andreas Strömbergsson, and Akshay Venkatesh.
Effective computation of maass cusp forms.
International mathematics research notices, 2006.

P. Deligne.
Formes modulaires et représentations l-adiques.
Séminaire N. Bourbaki, **355**:139–172, 1971.

Jay Jorgenson, Lejla Smajlović, and Holger Then.
On the distribution of eigenvalues of maass forms on certain moonshine groups.
Mathematics of Computation, 83(290):3039–3070, 2014.

References ii

H. Kim and P. Sarnak.
Refined estimates towards the Ramanujan and Selberg conjectures.
J. Amer. Math. Soc., **16**:175–181, 2003.

Holger Then.
Large sets of consecutive maass forms and fluctuations in the weyl remainder.
arXiv preprint arXiv:1212.3149, 2012.