Recent Results

Alex Karlovitz

In this document, I collect results from tests using the PARI code.

Tests for Hejhal's Algorithm

The code assumes we are working on a Hecke triangle group generated by $z \mapsto z+1$ and $z \mapsto -R^2/z$ for some $R > 0$. Here are a few values of R and corresponding ν values which are known to come from a true Maass form.

- $R = 1$, $\nu = i9.5336952613\ldots$
  - this is the group $SL(2, \mathbb{Z})$
- $R = 1/\sqrt{2}$, $\nu = i7.220872\ldots$
  - this is the congruence group Γ_4
- $R = 1/\sqrt{2}$, $\nu = i11.317680\ldots$
  - this is also the congruence group Γ_4, just a different eigenvalue
- $R = 7/20$, $\nu = 0.26705241700910205677150208864259506276668(7)\ldots$
  - note that this is an infinite volume fundamental domain, since $7/20 < 1/2$
  - note that ν is real in this example

July 2020

Tests Using all Three Expansions on Hecke Groups

Here, we collect results using the code in test_all_Hecke.pari. Specifically, we are running the function example_secant().

For each test, we list the input values, the output $s = \nu + 1/2$, and the approximate error from the true value.

Precision: until noted, using default(realprecision, 40) for all tests.

First, we want to vary which values are going into the secant method. Let's fix the following input.
Input:

$$r = 0.3500000000000000000000000000000$$

$$y_0 = 0.3000000000000000000000000000000 \quad M_1 = 10$$

$$\alpha_0 = 2.2000000000000000000000000000000 \quad M_2 = 10$$

$$\rho_0 = 3/4 \quad M_3 = 10$$

$$N_1 = 20 \quad N_2 = 20 \quad N_3 = 20$$

$$s_{start} = 0.76000000000000000000000000000000$$

$$\delta_{start} = 0.0050000000000000000000000000000000$$

Now we look at different outputs depending on our choices for the secant method.

Test using first two coefficients from cusp and flare expansions; output:

$$s_{guess} = 0.76705175510227623235618536383837$$

$$E \approx 6.6190682582441531672480422196285E-7$$

Test using first two coefficients from cusp expansion, plus first from flare and disk expansions; output:

$$s_{guess} = 0.76705218622576186081031266563593$$

$$E \approx 2.3078334019596118942300666740463E-7$$

Test using first two coefficients from each of the three expansions; output:

$$s_{guess} = 0.76705175510227623235618536383837$$

$$E \approx 6.6190682582441531672480422196285E-7$$

Test using first coefficient from each of the three expansions; output:

$$s_{guess} = 0.76705146318540003884924317275672$$

$$E \approx 9.5382370201792225891588587480777E-7$$

From now on, tests use the first two coefficients from the cusp expansion, and the first coefficient from the flare and disk expansions. As seen above, this is the method which performed best on the given example.

Input:

$$r = 0.3500000000000000000000000000000000000000$$

$$y_0 = 0.3100000000000000000000000000000000000000 \quad M_1 = 20$$

$$\alpha_0 = 2.300000000000000000000000000000000000000 \quad M_2 = 12$$

$$\rho_0 = 3/4 \quad M_3 = 5$$

$$N_1 = 22 \quad N_2 = 24 \quad N_3 = 6$$

$$s_{start} = 0.7670524170000000000000000000000000000000$$

$$\delta_{start} = 2.000000000000000000000000000000000000000E-11$$

Output:

$$s_{guess} = 0.7670921794552911637659592720972066442132$$

$$E \approx 3.976244618910699445718345461158144652910E-5$$

Input:

$$r = 0.3500000000000000000000000000000000000000$$

$$y_0 = 0.3100000000000000000000000000000000000000 \quad M_1 = 20$$

$$\alpha_0 = 2.300000000000000000000000000000000000000 \quad M_2 = 12$$

$$\rho_0 = 3/4 \quad M_3 = 5$$

$$N_1 = 22 \quad N_2 = 24 \quad N_3 = 0$$

$$s_{start} = 0.7670524170000000000000000000000000000000$$

$$\delta_{start} = 2.000000000000000000000000000000000000000E-11$$

Output:

$$s_{guess} = 0.76711244060191816956972608596719150655 28$$

$$E \approx 6.0023592816112798223997324596443786130 82E-5$$

Input:

$$r = 0.350000000000000000000000000000000000000$$

$$y_0 = 0.310000000000000000000000000000000000000 \quad M_1 = 20$$

$$\alpha_0 = 2.300000000000000000000000000000000000000 \quad M_2 = 12$$

$$\rho_0 = 3/4 \quad M_3 = 12$$

$$N_1 = 22 \quad N_2 = 24 \quad N_3 = 0$$

$$s_{start} = 0.767052417000000000000000000000000000000$$

$$\delta_{start} = 2.000000000000000000000000000000000000000E-11$$

Output:

$$s_{guess} = 0.767052417033854813534261176221014388 3645$$

$$E \approx 2.475275676275908757841932559781421248042E-11$$

Input:

$$r = 0.350000000000000000000000000000000000000$$

$$y_0 = 0.310000000000000000000000000000000000000 \quad M_1 = 20$$

$$\alpha_0 = 2.300000000000000000000000000000000000000 \quad M_2 = 12$$

$$\rho_0 = 3/4 \quad M_3 = 20$$

$$N_1 = 22 \quad N_2 = 24 \quad N_3 = 0$$

$$s_{start} = 0.767052417000000000000000000000000000000$$

$$\delta_{start} = 2.000000000000000000000000000000000000000E-11$$

Output:

$$s_{guess} = 0.767052417009102051293377443417012463 9491$$

$$E \approx 5.478124645225582598817609789207250709646E-18$$

Input:

$$r = 0.350000000000000000000000000000000000000$$

$$y_0 = 0.310000000000000000000000000000000000000 \quad M_1 = 20$$

$$\alpha_0 = 2.300000000000000000000000000000000000000 \quad M_2 = 12$$

$$\rho_0 = 3/4 \quad M_3 = 20$$

$$N_1 = 22 \quad N_2 = 24 \quad N_3 = 6$$

$$s_{start} = 0.767052417000000000000000000000000000000$$

$$\delta_{start} = 2.000000000000000000000000000000000000000E-11$$

Output:

$$s_{guess} = 0.767052417009101823893719489865012069 1583$$

$$E \approx 2.328777825987775829936083804947119040820E-16$$

Thoughts: the above slew of tests suggests two things.

1. Including a relatively high number of coefficients in the disk expansion (M_3) seems to be necessary.

2. Taking test points from the disk model (N_3) seems to actually *hurt* performance. Why are these points not useful (and even further, damaging)?

Input:

$$r = 0.35000000000000000000000000000000000000$$

$$y_0 = 0.31000000000000000000000000000000000000 \quad M_1 = 20$$

$$\alpha_0 = 2.3000000000000000000000000000000000000 \quad M_2 = 12$$

$$\rho_0 = 3/4 \quad M_3 = 30$$

$$N_1 = 22 \quad N_2 = 24 \quad N_3 = 0$$

$$s_{start} = 0.76705241700000000000000000000000000000$$

$$\delta_{start} = 2.0000000000000000000000000000000000000E-11$$

Output:

$$s_{guess} = 0.76705241700910205612288725661764384474$$

$$E \approx 6.4861483202495121802575748064887076918E-19$$

Okay, so we have s to about 19 decimal places. Let's start there and try to zoom in.

Input:

$$r = 0.35000000000000000000000000000000000000$$

$$y_0 = 0.31000000000000000000000000000000000000 \quad M_1 = 26$$

$$\alpha_0 = 2.3000000000000000000000000000000000000 \quad M_2 = 16$$

$$\rho_0 = 3/4 \quad M_3 = 26$$

$$N_1 = 28 \quad N_2 = 32 \quad N_3 = 0$$

$$s_{start} = 0.76705241700910205670000000000000000000$$

$$\delta_{start} = 3.0000000000000000000000000000000000000E-19$$

Output:

$$s_{guess} = 0.7670524170091020567$$

$$E \approx 0.E-19$$

Note: obviously some kind of coding issue came up in the previous test. Maybe if we increase precision to default(realprecision, 60)?

Precision: until noted, using default(realprecision, 60) for all tests below.

Input:

$$r = 0.350000000000000000000000000000000000000000000000000000000$$

$$y_0 = 0.310000000000000000000000000000000000000000000000000000000 \quad M_1 = 26$$

$$\alpha_0 = 2.30000000000000000000000000000000000000000000000000000000 \quad M_2 = 16$$

$$\rho_0 = 3/4 \quad M_3 = 26$$

$$N_1 = 28 \quad N_2 = 32 \quad N_3 = 0$$

$$s_{start} = 0.767052417009102056700000000000000000000000000000000000000$$

$$\delta_{start} = 3.00000000000000000000000000000000000000000000000000000000E-19$$

Output:

$$s_{guess} = 0.76705241700910205677316120102884709893$$

$$E \approx 1.65911238625203616543E - 21$$

Input:

$$r = 0.350000000000000000000000000000000000000000000000000000000000$$

$$y_0 = 0.310000000000000000000000000000000000000000000000000000000000 \quad M_1 = 26$$

$$\alpha_0 = 2.30000000000000000000000000000000000000000000000000000000000 \quad M_2 = 16$$

$$\rho_0 = 3/4 \quad M_3 = 36$$

$$N_1 = 28 \quad N_2 = 32 \quad N_3 = 0$$

$$s_{start} = 0.767052417009102056700000000000000000000000000000000000000000$$

$$\delta_{start} = 3.00000000000000000000000000000000000000000000000000000000000E - 19$$

Output:

$$s_{guess} = 0.76705241700910205677150275933072804301$$

$$E \approx 6.706881329802452983E - 25$$

Input:

$$r = 0.350000000000000000000000000000000000000000000000000000000000$$

$$y_0 = 0.310000000000000000000000000000000000000000000000000000000000 \quad M_1 = 30$$

$$\alpha_0 = 2.30000000000000000000000000000000000000000000000000000000000 \quad M_2 = 20$$

$$\rho_0 = 3/4 \quad M_3 = 40$$

$$N_1 = 32 \quad N_2 = 40 \quad N_3 = 0$$

$$s_{start} = 0.767052417009102056700000000000000000000000000000000000000000$$

$$\delta_{start} = 3.00000000000000000000000000000000000000000000000000000000000E - 19$$

Output:

$$s_{guess} = 0.76705241700910205677150208864814423283$$

$$E \approx 5.549170067453791727E - 30$$

Alright, let's see how close we can get to the full number of digits.

Note: I had to increase PARI's stack size to be able to handle the computations below. Each test only took a few minutes, but clearly required a lot of RAM.

Input:

$$r = 0.350000000000000000000000000000000000000000000000000000000000$$

$$y_0 = 0.310000000000000000000000000000000000000000000000000000000000 \quad M_1 = 60$$

$$\alpha_0 = 2.30000000000000000000000000000000000000000000000000000000000 \quad M_2 = 35$$

$$\rho_0 = 3/4 \quad M_3 = 60$$

$$N_1 = 62 \quad N_2 = 70 \quad N_3 = 0$$

$$s_{start} = 0.767052417009102056771502088642595062500000000000000000000000$$

$$\delta_{start} = 4.00000000000000000000000000000000000000000000000000000000000E - 37$$

Output:

$$s_{guess} = 0.76705241700910205677150208864259506268201354661690277139O8$$

$$E \approx 8.4666453383097228609049572158984559319E - 38$$

Couldn't do better than 38 digits until I increased the precision again.

Precision: until noted, using default(realprecision, 80) for all tests below.

Input:

$$r = 0.35000000000000000000000000000000000000000000000000000000000000000000000000000000$$

$$y_0 = 0.31000000000000000000000000000000000000000000000000000000000000000000000000000000 \quad M_1 = 65$$

$$\alpha_0 = 2.3000000000000000000000000000000000000000000000000000000000000000000000000000000 \quad M_2 = 40$$

$$\rho_0 = 3/4 \quad M_3 = 75$$

$$N_1 = 67 \quad N_2 = 80 \quad N_3 = 0$$

$$s_{start} = 0.76705241700910205677150208864259506250000000000000000000000000000000000000000000$$

$$\delta_{start} = 4.0000000000000000000000000000000000000000000000000000000000000000000000000000000E - 37$$

Output:

$$s_{guess} = 0.7670524170091020567715020886425950627666870757682116254668$$

$$E \approx 7.075768211625466915E - 42$$

We tied Strömbergsson's precision! Now we need to print the secant method steps and see if we think we're doing better.

Precision: went up to default(realprecision, 100) for the computations below.
Input:

$$r = 0.35$$

$$y_0 = 0.31 \quad M_1 = 75$$

$$\alpha_0 = 2.3 \quad M_2 = 50$$

$$\rho_0 = 3/4 \quad M_3 = 85$$

$$N_1 = 77 \quad N_2 = 100 \quad N_3 = 0$$

$$s_{start} = 0.7670524170091020567715020886425950625$$

$$\delta_{start} = 4.0E - 37$$

Output:

$$s_{guess} = 0.767052417009102056771502088642595062766687077618143794789163072259008589312 00$$

$$E \approx 7.077618143794789163072259008589312 0048E - 42$$

Final set of predictions:

$$0.76705241700910205677150208864259506276668707761814367412713281055516159638721$$

$$0.76705241700910205677150208864259506276668707761814375238526869680606787332718$$

$$0.76705241700910205677150208864259506276668707761814382936130083054086675095072$$

$$0.76705241700910205677150208864259506276668707761814391545119333396285558223679$$

This seems to suggest the true eigenvalue lies between

$$0.767052417009102056771502088642595062766687077618 1437 \pm 1e - 52$$

That is, we appear to have found 51 decimal places. How can we check this?

Tests Using just Cusp and Flare (Strömbergsson's Code)

Running

```
default(realprecision,80);
r=0.35; alpha=2.2; y0=0.34; M=60; MM=35;
zoomin(r,alpha,y0,M,MM,0.7670524170091020567715020886425950625,4e-37);
```

results in

```
New predictions:
7.67052417009102056771502088642595062766687077618142806858366156855094778791353590172130106595438 E-1
7.67052417009102056771502088642595062766687077618143235168970024034342605904962851352719017652213 E-1
7.67052417009102056771502088642595062766687077618142234036889656789028573116993374108850338642172 E-1
7.67052417009102056771502088642595062766687077618141017766554129340601157752762344131337905826968 E-1
Approximate error:2.217402415894693741448152200507221381111825245393746583451 E-51
```

So we try starting closer to that value. Running

```
default(realprecision,80);
r=0.35; alpha=2.2; y0=0.34; M=80; MM=45;
zoomin(r,alpha,y0,M,MM,0.767052417009102056771502088642595062766687077618,4e-47);
```

results in

```
New predictions:
7.67052417009102056771502088642595062766687077618142701698223967688582627703507519701403781776194 E-1
7.67052417009102056771502088642595062766687077618142701698223967689282128903537979165535461450436 E-1
7.67052417009102056771502088642595062766687077618142701698223967661977542295045586530478342260844 E-1
7.67052417009102056771502088642595062766687077618142701698223967697910545076894113387573752706499 E-1
Approximate error:3.59330027818485268570954104456546927338 E-65
```

And more precise:

```
default(realprecision,100);
r=0.35; alpha=2.2; y0=0.34; M=100; MM=65;
zoomin(r,alpha,y0,M,MM,0.76705241700910205677150208864259506276668707761814270169822396,4e-62);
```

results in

```
New predictions:
7.6705241700910205677150208864259506276668707761814270169822396768661164742932102542... E-1
7.6705241700910205677150208864259506276668707761814270169822396768661225697612949350... E-1
7.6705241700910205677150208864259506276668707761814270169822396768661147230540627548... E-1
7.6705241700910205677150208864259506276668707761814270169822396768661021492553701202... E-1
Approximate error:2.042050592481475195878038352430303093200589584563292333435 E-69
```

March-April 2020

We want to see how many digits of precision the secant method can achieve in the infinite volume case. We expect that varying the number of Fourier coefficients used in the two expansions as well as the internal precision PARI will lead to different results.

In both methods below, we write M_1 for the number of Fourier coefficients taken in the cuspidal expansion and M_2 for the number of coefficients in the flare expansion. Values in the tables are obtained using the stated method with the parameters

$$r = \frac{7}{20} \quad \alpha = 2 \quad y_0 = 0.32 \quad s_{\text{start}} = 0.76$$

Secant Method Results

We used the parameter

$$\delta_{\text{start}} = 0.01$$

for the secant method.

M_1	M_2	PARI precision	Digits correct
15	5	32	~ 8
15	5	64	~ 8
15	10	32	~ 12
15	15	32	~ 12
15	20	32	~ 12
20	5	32	~ 8
20	10	32	~ 16
20	15	32	~ 17
20	20	32	~ 17
25	5	32	~ 9
25	10	32	~ 17
25	15	32	~ 19
25	15	64	~ 19
25	15	128	~ 19
25	15	256	~ 19
25	15	512	~ 19
25	20	32	~ 18
30	5	32	~ 8
30	10	32	~ 16
30	15	32	~ 19
30	20	32	~ 18
35	15	32	~ 19
40	15	32	~ 18
45	15	32	~ 17
50	15	32	~ 18
50	20	32	~ 19
50	20	64	~ 19
80	50	32	~ 2
80	50	64	~ 2

Grid Method Results

We used **FILL IN** grid points for the grid method.

M_1	M_2	PARI precision	Digits correct
25	15	32	~ 19