Computational Aspects of Maass Waveforms

Fredrik Strömberg

Dept. of Math. Uppsala University

15 March 2005
Main Accomplishments

- A general method to compute Maass waveforms:
Main Accomplishments

- A general method to compute Maass waveforms:
  - For any cofinite Fuchsian group with cusps.
Main Accomplishments

- A general method to compute Maass waveforms:
  - For any cofinite Fuchsian group with cusps.
  - For any multiplier system $\nu$, and any real weight $k$. 
Main Accomplishments

- A general method to compute Maass waveforms:
  - For any cofinite Fuchsian group with cusps.
  - For any multiplier system \( \nu \), and any real weight \( k \).

Most Important Experimental Results
Most Important Experimental Results

- Weight $k \to 0$
Most Important Experimental Results

- Weight $k \to 0$
  - Individual convergence to cusp forms/Eisenstein series.
Most Important Experimental Results

- Weight $k \to 0$
  - Individual convergence to cusp forms/Eisenstein series.
  - Evenly spaced $R$’s.
Most Important Experimental Results

- Weight $k \rightarrow 0$
  - Individual convergence to cusp forms/Eisenstein series.
  - Evenly spaced $R$’s.

- Non-congruence subgroups
Most Important Experimental Results

- Weight $k \to 0$
  - Individual convergence to cusp forms/Eisenstein series.
  - Evenly spaced $R$’s.

- Non-congruence subgroups
  - Phillips-Sarnak conjecture.
Most Important Experimental Results

- Weight $k \to 0$
  - Individual convergence to cusp forms/Eisenstein series.
  - Evenly spaced $R$’s.

- Non-congruence subgroups
  - Phillips-Sarnak conjecture.
  - Fourier coefficients with different Gaussians.
Most Important Experimental Results

▶ Weight $k \to 0$
  ▶ Individual convergence to cusp forms/Eisenstein series.
  ▶ Evenly spaced $R$’s.

▶ Non-congruence subgroups
  ▶ Phillips-Sarnak conjecture.
  ▶ Fourier coefficients with different Gaussians.

▶ Weight $k = \frac{1}{2}$
Most Important Experimental Results

- Weight \( k \to 0 \)
  - Individual convergence to cusp forms/Eisenstein series.
  - Evenly spaced \( R \)'s.

- Non-congruence subgroups
  - Phillips-Sarnak conjecture.
  - Fourier coefficients with different Gaussians.

- Weight \( k = \frac{1}{2} \)
  - Shimura correspondence for Maass waveforms:
    \((\Gamma_0(2), \text{weight}=0) \leftrightarrow (\Gamma_0(4), \text{weight}=\frac{1}{2})\)
Most Important Experimental Results

- **Weight** $k \rightarrow 0$
  - Individual convergence to cusp forms/Eisenstein series.
  - Evenly spaced $R$’s.

- **Non-congruence subgroups**
  - Phillips-Sarnak conjecture.
  - Fourier coefficients with different Gaussians.

- **Weight** $k = \frac{1}{2}$
  - Shimura correspondence for Maass waveforms: $(\Gamma_0(2), \text{weight}=0) \leftrightarrow (\Gamma_0(4), \text{weight}=\frac{1}{2})$
  - Distribution of Fourier coefficients $a(t), t$ square-free
Original problem

We want solutions of:

\[
\Delta \phi + \lambda \phi = 0, \\
\phi(Tz) = \phi(z), \forall T \in \Gamma, \\
\int_{\Gamma \backslash \mathcal{H}} |\phi|^2 d\mu < \infty,
\]

where \(\Delta \phi = y^2 \left( \phi''_{xx} + \phi''_{yy} \right)\), and in the original setting \(\Gamma = PSL(2, \mathbb{Z})\).
For $\lambda = \frac{1}{4} + R^2$ the solutions can be written as

$$
\phi(x + iy) = \sum_{n=-\infty}^{\infty} c(n) \sqrt{Y} K_{iR} (2\pi |n| y) e^{2\pi inx}.
$$

We want to compute the Fourier coefficients $c(n)$ (usually $c(1) = 1$).

I.e. a set of complex numbers $\{c(n)\}$ such that the linear combination above satisfy

$$
\phi(Tz) = \phi(z), \forall T \in \Gamma.
$$
Stark’s method

- Hecke Operators $T_p(\phi)(z) = c(p)\phi(z)$. 

- Non-linear system, iterative techniques to find fixed-point

- Not robust! (for large $R$)

- Only where there are Hecke operators!
Stark’s method

- Hecke Operators $T_p(\phi)(z) = c(p)\phi(z)$.
- Non-linear system, iterative techniques to find fixed-point

$$T(\vec{c}) = \vec{c}.$$
Stark’s method

- Hecke Operators $T_p(\phi)(z) = c(p)\phi(z)$.
- Non-linear system, iterative techniques to find fixed-point $T(\vec{c}) = \vec{c}$.
- Not robust! (for large $R$)
Stark’s method

- Hecke Operators $T_p (\phi) (z) = c(p)\phi(z)$.
- Non-linear system, iterative techniques to find fixed-point $T(\vec{c}) = \vec{c}$.
- Not robust! (for large $R$)
- Only where there are Hecke operators!
Hejhal’s method for one cusp groups

- Finite Fourier series $\phi = \hat{\phi} + [[\varepsilon]]$ (treat $[[\varepsilon]]$ as 0): 
  $$\hat{\phi}(x + iy) = \sum_{1 \leq |n| \leq M(y)} c(n) \sqrt{y} K_{iR} (2\pi |n| y) e^{2\pi i nx}.$$
Hejhal’s method for one cusp groups

- Finite Fourier series $\phi = \hat{\phi} + [[\varepsilon]]$ (treat $[[\varepsilon]]$ as 0):
  \[
  \hat{\phi}(x + iy) = \sum_{1 \leq |n| \leq M(y)} c(n) \sqrt{y} K_{iR} (2\pi|n|y) e^{2\pi inx}.
  \]

- Invert using $z_m = \frac{1}{2Q} (m - \frac{1}{2}) + iY$, (for $|n| \leq M(Y) < Q$)
  \[
  c(n) \sqrt{Y} K_{iR} (2\pi|n|Y) = \frac{1}{2Q} \sum_{m=1-Q}^{Q} \hat{\phi}(z_m) e^{-2\pi inx_m}
  \]
Hejhal’s method for one cusp groups

- Finite Fourier series $\phi = \hat{\phi} + [[\varepsilon]]$ (treat $[[\varepsilon]]$ as 0):
  $$\hat{\phi}(x + iy) = \sum_{1 \leq |n| \leq M(y)} c(n) \sqrt{y} K_{iR}(2\pi |n| y) e^{2\pi inx}.$$  

- Invert using $z_m = \frac{1}{2Q} (m - \frac{1}{2}) + iY$, (for $|n| \leq M(Y) < Q$)
  $$c(n) \sqrt{Y} K_{iR}(2\pi |n| Y) = \frac{1}{2Q} \sum_{m=1}^{Q} \hat{\phi}(z_m) e^{-2\pi inx_m}$$

- Implicit automorphy: $\hat{\phi}(z_m) = \hat{\phi}(z_m^*), z_m^* = $ pullback of $z_m$
Hejhal’s method for one cusp groups

- Finite Fourier series $\phi = \hat{\phi} + [[\varepsilon]]$ (treat $[[\varepsilon]]$ as 0):
  $$\hat{\phi}(x + iy) = \sum_{1 \leq |n| \leq M(y)} c(n) \sqrt{y} K_{iR} (2\pi |n| y) e^{2\pi i n x}.$$

- Invert using $z_m = \frac{1}{2Q} \left( m - \frac{1}{2} \right) + iY$, (for $|n| \leq M(Y) < Q$)
  $$c(n) \sqrt{Y} K_{iR} (2\pi |n| Y) = \frac{1}{2Q} \sum_{m=1-Q}^{Q} \hat{\phi}(z_m) e^{-2\pi i n x_m}.$$

- Implicit automorphy: $\hat{\phi}(z_m) = \hat{\phi}(z_m^*), z_m^* = $ pullback of $z_m$

- Let $M_0 = M(Y_{min})$. 
Hejhal’s method for one cusp groups

- Finite Fourier series $\phi = \hat{\phi} + [[\varepsilon]]$ (treat $[[\varepsilon]]$ as 0):
  $$\hat{\phi}(x + iy) = \sum_{1 \leq |n| \leq M(y)} c(n) \sqrt{y} K_{iR}(2\pi |n| y) e^{2\pi i nx}.$$  

- Invert using $z_m = \frac{1}{2Q} (m - \frac{1}{2}) + i Y$, (for $|n| \leq M(Y) < Q$)
  $$c(n) \sqrt{Y} K_{iR}(2\pi |n| Y) = \frac{1}{2Q} \sum_{m=1-Q}^{Q} \hat{\phi}(z_m) e^{-2\pi i nx_m}.$$  

- Implicit automorphy: $\hat{\phi}(z_m) = \hat{\phi}(z^*_m)$, $z^*_m = \text{pullback of } z_m$

- Let $M_0 = M(Y_{\text{min}})$.

- Well-conditioned linear system for $\vec{c} = (c(-M_0), \ldots, c(M_0))$:
  $$V \vec{c} = \vec{0}.$$
Hejhal’s method for one cusp groups

- Finite Fourier series $\phi = \hat{\phi} + [[\varepsilon]]$ (treat $[[\varepsilon]]$ as 0):
  $$\hat{\phi}(x + iy) = \sum_{1 \leq |n| \leq M(y)} c(n) \sqrt{y} K_{iR}(2\pi|n|y) e^{2\pi i nx}.$$

- Invert using $z_m = \frac{1}{2Q} (m - \frac{1}{2}) + iY$, (for $|n| \leq M(Y) < Q$)
  $$c(n) \sqrt{Y} K_{iR}(2\pi|n|Y) = \frac{1}{2Q} \sum_{m=1}^{Q} \hat{\phi}(z_m) e^{-2\pi i nx_m}.$$

- Implicit automorphy: $\hat{\phi}(z_m) = \hat{\phi}(z_m^*)$, $z_m^*$ = pullback of $z_m$

- Let $M_0 = M(Y_{min})$.

- Well-conditioned linear system for $\vec{c} = (c(-M_0), \ldots, c(M_0))$:
  $$V \vec{c} = \vec{0}.$$

- Solved by standard linear algebra techniques.
Hejhal’s method for one cusp groups

- Finite Fourier series $\phi = \hat{\phi} + [\epsilon]$ (treat $[\epsilon]$ as 0):
  $$\hat{\phi}(x + iy) = \sum_{1 \leq |n| \leq M(y)} c(n) \sqrt{y} K_{iR} (2\pi |n| y) e^{2\pi inx}.$$ 

- Invert using $z_m = \frac{1}{2Q} \left( m - \frac{1}{2} \right) + iY$, (for $|n| \leq M(Y) < Q$)
  $$c(n) \sqrt{Y} K_{iR} (2\pi |n| Y) = \frac{1}{2Q} \sum_{m=1-Q}^{Q} \hat{\phi} (z_m) e^{-2\pi inx_m}.$$

- Implicit automorphy: $\hat{\phi}(z_m) = \hat{\phi} (z_m^*)$, $z_m^*=$ pullback of $z_m$

- Let $M_0 = M(Y_{\min})$.

- Well-conditioned linear system for $\vec{c} = (c(-M_0), \ldots, c(M_0))$:
  $$V\vec{c} = \vec{0}.$$

- Solved by standard linear algebra techniques.

- Robust! (H. Then: $R > 40000$)
The key point

Note that if the $z_m$'s are all below the fundamental domain, i.e. $Y < Y_{min}$, then in

$$c(n) \sqrt{Y} K_{iR}(2\pi|n|Y) = \frac{1}{2Q} \sum_{m=1-Q}^{Q} \hat{\phi}(z_m^*) e^{-2\pi i nx_m},$$  

we can truncate all $\hat{\phi}(z_m^*)$ at the same point, $M_0$. Hence the right hand side contains only the coefficients $c(-M_0), \ldots, c(M_0)$.

Phase 1: We solve the linear system for the coefficients $c(-M_0), \ldots, c(M_0)$.

Phase 2: Using (*) we can solve for any $c(n)$, $|n| > M_0$ in terms of the first $2M_0$ coefficients!
How do we find eigenvalues (i.e. $R$)?

The solution $\vec{c} = \vec{c}(R, Y)$ is

- continuous in $R$ and
- for a true eigenvalue, independent of $Y$.

We solve $V\vec{c} = \vec{0}$ for two $Y$'s in parallel, giving $\vec{c}$ and $\vec{c}'$ and form a functional like

$$h(R) = |c(2) - c'(2)| + |c(3) - c'(3)| + |c(4) - c'(4)|,$$

which is minimized over a grid in $R$. 
General problem

\[ \Gamma = \text{cofinite with } \kappa \geq 1 \text{ cusps}, \; k \in \mathbb{R}, \; \nu : \overline{\Gamma} \to S^1 \text{ a multiplier system.} \]
General problem

\[ \Gamma = \text{cofinite with } \kappa \geq 1 \text{ cusps, } k \in \mathbb{R}, \nu : \overline{\Gamma} \to S^1 \text{ a multiplier system.} \]

We have to replace the Laplacian with the \( k \)-Laplacian:

\[ \Delta_k = \Delta - iyk \frac{\partial}{\partial x}, \]
General problem

$\Gamma = \text{cofinite with } \kappa \geq 1 \text{ cusps, } k \in \mathbb{R}, \nu : \overline{\Gamma} \to S^1 \text{ a multiplier system.}$

We have to replace the Laplacian with the $k$-Laplacian:

$$\Delta_k = \Delta - iyk \frac{\partial}{\partial x},$$

and the invariance property with

$$\phi(Tz) = j_T(z; k) \nu(T) \phi(z), \forall T \in \Gamma,$$

where $j_T(z; k) = e^{ik \text{Arg}(cz+d)}$, for $T = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. 
A General Pullback

General pullback procedure needs either
generators, or coset representatives.
Virtually no impact on the total CPU-time.
A General Pullback

- General pullback procedure needs either
  - generators, or
A General Pullback

- General pullback procedure needs either
  - generators, or
  - coset representatives.
A General Pullback

- General pullback procedure needs either
  - generators, or
  - coset representatives.

- Virtually no impact on the total CPU-time.
Several Fourier Series

For numerical stability we must use one Fourier series connected to each cusp:

\[ \phi_j(z) = \phi(\sigma_j z) = \sum_{|n| \geq 1} c_j(n) K_{iR}(2\pi|n|y) e(nx), \]

where \( \sigma_j^{-1} \) maps the cuspidal region to the standard form at \( i\infty \).

- Increasing the size of the linear system!
- Moderate impact on the total CPU time.
Generalized Automorphy

Note that the automorphy relation $\hat{\phi}(z_m) = \hat{\phi}(z_m^*)$ is replaced by

$$\hat{\phi}(z_m) = \hat{\phi}(z_m^*) v(T_m)^{-1} j_{T_m} (z_m; k)^{-1}.$$ 

Phase 1: $|n| \leq "M_0"$ (the analogue of $M_0$)

Phase 2: $|n| > "M_0".$
Whittaker Functions

When we replaced $\Delta$ by $\Delta_k$ we also have to replace the K-Bessel function with the Whittaker $W$-function in the Fourier series:

$$K_{iR}(x) \rightarrow W_{\pm \frac{1}{2}k,iR}(2x)$$

This have a huge impact on the CPU-time. A factor of 10-100.
The time it takes to search for Maass waveforms (on 3.3GHz CPU):

- $\Gamma_0(2), k = 0, R \leq 20$: 40 seconds ($\sim 40$).

- $\Gamma_0(2), k = 1, R \leq 20$: 115min ($\sim 95$)
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involutions (preserving the given space of Maass waveforms)

$\Rightarrow$ $c_j(n)$ are real

$\Rightarrow$ Reflection

Atkin-Lehner/Fricke involutions

At non-zero weight: Conjugation & Reflection

$\Rightarrow$ $c_j(n)$ are real.

Hecke operators: Multiplicativity

$\Rightarrow$ $c_1(2)c_1(3) = c_1(6)$ (for example)

Some explicit formulas for eigenvalues and coefficients (i.e. CM-forms)!

In most cases these tests indicate between 10-12 correct digits.

I would stake my life on these numbers! (worst Hecke relation -1 to be on the safe side)
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involutions (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n)$ are real
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involutions (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n)$ are real
  - Reflection $\Rightarrow c_j(-n) = \varepsilon_j c_j(n)$,
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involution (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n)$ are real
  - Reflection $\Rightarrow c_j(-n) = \varepsilon_j c_j(n)$
  - Atkin-Lehner/Fricke involutions $\Rightarrow c_j(n) = \mu_j c_1(n)$

In most cases these tests indicate between 10-12 correct digits. I would stake my life on these numbers! (worst Hecke relation -1 to be on the safe side)
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involutions (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n)$ are real
  - Reflection $\Rightarrow c_j(-n) = \varepsilon_j c_j(n)$,
  - Atkin-Lehner/Fricke involutions $\Rightarrow c_j(n) = \mu_j c_1(n)$
  - At non-zero weight: Conjugation & Reflection $\Rightarrow c_j(n)$ are real.

In most cases these tests indicate between 10-12 correct digits. I would stake my life on these numbers! (worst Hecke relation -1 to be on the safe side)
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involutions (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n)$ are real
  - Reflection $\Rightarrow c_j(-n) = \varepsilon_j c_j(n)$,
  - Atkin-Lehner/Fricke involutions $\Rightarrow c_j(n) = \mu_j c_1(n)$
  - At non-zero weight: Conjugation & Reflection $\Rightarrow c_j(n)$ are real.
- Hecke operators: Multiplicativity $\Rightarrow c_1(2)c_1(3) = c_1(6)$ (for example)
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involutions (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n) \text{ are real}$
  - Reflection $\Rightarrow c_j(-n) = \varepsilon_j c_j(n)$
  - Atkin-Lehner/Fricke involutions $\Rightarrow c_j(n) = \mu_j c_1(n)$
  - At non-zero weight: Conjugation & Reflection $\Rightarrow c_j(n) \text{ are real}$.
- Hecke operators: Multiplicativity $\Rightarrow c_1(2)c_1(3) = c_1(6)$ (for example)
- Some explicit formulas for eigenvalues and coefficients (i.e. CM-forms)!
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c'_j(k)$
- Involutions (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n)$ are real
  - Reflection $\Rightarrow c_j(-n) = \varepsilon_j c_j(n)$
  - Atkin-Lehner/Fricke involutions $\Rightarrow c_j(n) = \mu_j c_1(n)$
  - At non-zero weight: Conjugation & Reflection $\Rightarrow c_j(n)$ are real.
- Hecke operators: Multiplicativity $\Rightarrow c_1(2)c_1(3) = c_1(6)$ (for example)
- Some explicit formulas for eigenvalues and coefficients (i.e. CM-forms)!

In most cases these tests indicate between 10-12 correct digits.
Measures of Accuracy for $R$ and $c(n)$

We use properties which are not built in to the algorithm:

- Multiple independent choices of $Y$: $c_j(k) = c_j'(k)$
- Involution (preserving the given space of Maass waveforms)
  - Complex conjugation $\Rightarrow c_j(n)$ are real
  - Reflection $\Rightarrow c_j(-n) = \varepsilon_j c_j(n)$
  - Atkin-Lehner/Fricke involutions $\Rightarrow c_j(n) = \mu_j c_1(n)$
  - At non-zero weight: Conjugation & Reflection $\Rightarrow c_j(n)$ are real.
- Hecke operators: Multiplicativity $\Rightarrow c_1(2)c_1(3) = c_1(6)$ (for example)
- Some explicit formulas for eigenvalues and coefficients (i.e. CM-forms)!

In most cases these tests indicate between 10-12 correct digits.
I would stake my life on these numbers! (worst Hecke relation -1 to be on the safe side)
Previous Attempt at $k \neq 0$

- Main problems: Inaccurate Whittaker function and ill-conditioned system.
Results I am most fond of

There are four experimental observations in particular that I like

1. Multiplicity 2 for real Dirichlet characters.
Results I am most fond of

There are four experimental observations in particular that I like

1. Multiplicity 2 for real Dirichlet characters.
2. Individual convergence as $k \to 0$. 
Results I am most fond of

There are four experimental observations in particular that I like

1. Multiplicity 2 for real Dirichlet characters.
2. Individual convergence as $k \to 0$.
3. Lift at weight 1 from $\Gamma_0(1)$ and eta-multiplier to $\Gamma_0(144)$ and Dirichlet character.
Results I am most fond of

There are four experimental observations in particular that I like

1. Multiplicity 2 for real Dirichlet characters.
2. Individual convergence as $k \to 0$.
3. Lift at weight 1 from $\Gamma_0(1)$ and eta-multiplier to $\Gamma_0(144)$ and Dirichlet character.
4. Distribution of Fourier coefficients for non-congruence subgroups.
Some Future Applications of the Algorithm

1. Extend Hejhal's work on value distributions to these new cases.
Some Future Applications of the Algorithm

1. Extend Hejhal's work on value distributions to these new cases.
2. Exceptional eigenvalues ($\lambda < \frac{1}{4}$) on non-congruence subgroups.
Some Future Applications of the Algorithm

1. Extend Hejhal’s work on value distributions to these new cases.
2. Exceptional eigenvalues ($\lambda < \frac{1}{4}$) on non-congruence subgroups.
3. Statistics for the lowest eigenvalues (as the group varies).
Some Future Applications of the Algorithm

1. Extend Hejhal's work on value distributions to these new cases.
2. Exceptional eigenvalues \((\lambda < \frac{1}{4})\) on non-congruence subgroups.
3. Statistics for the lowest eigenvalues (as the group varies).
4. Non-arithmetic characters.
Some Future Applications of the Algorithm

1. Extend Hejhal's work on value distributions to these new cases.
2. Exceptional eigenvalues ($\lambda < \frac{1}{4}$) on non-congruence subgroups.
3. Statistics for the lowest eigenvalues (as the group varies).
4. Non-arithmetic characters.
5. More general groups (not subgroups of $PSL(2, \mathbb{Z})$).
Open Theoretical Problems

Most interesting (involved):

1. Prove the individual convergence as $k \to 0$. 
Open Theoretical Problems

Most interesting (involved):

1. Prove the individual convergence as $k \to 0$.
2. For noncongruence subgroups prove (even heuristically) that coefficients can have different s.d. for different congruence classes.
Open Theoretical Problems

Most interesting (involved):

1. Prove the individual convergence as $k \to 0$.
2. For noncongruence subgroups prove (even heuristically) that coefficients can have different s.d. for different congruence classes.
Open Theoretical Problems

Most interesting (involved):

1. Prove the individual convergence as $k \to 0$.
2. For noncongruence subgroups prove (even heuristically) that coefficients can have different s.d. for different congruence classes.

Smaller open problems:

1. Prove a detailed version of Weyl’s law for $\Gamma_0(N)$ and $\chi \neq 1$. 
Open Theoretical Problems

Most interesting (involved):

1. Prove the individual convergence as $k \to 0$.
2. For noncongruence subgroups prove (even heuristically) that coefficients can have different s.d. for different congruence classes.

Smaller open problems:

1. Prove a detailed version of Weyl’s law for $\Gamma_0(N)$ and $\chi \neq 1$.
2. Prove a detailed version of Weyl’s law as $k \to 0$ (for $R$ bounded).
Physical Applications

- Applications of Maass Waveforms
  - Quantum Chaos
  - Cosmology (cf. Holger Then)

- Related Areas
  - Quantum wires/dots and devices. (cf. the books by Norm Hurt)
Multiplicity 2

Setting: $\Gamma = \Gamma_0(p), \chi = \left( \frac{.}{p} \right)$.

Experimental observation: $f_+$ and $f_-$ are two different functions with the same eigenvalue.
### Multiplicity 2

Setting: $\Gamma = \Gamma_0(p)$, $\chi = \left( \frac{.}{p} \right)$.

**Experimental observation:** $f_+$ and $f_-$ are two different functions with the same eigenvalue.

**Conclusion:** For real even Dirichlet character, in general the spectrum have multiplicity 2.
### Multiplicity 2

**Setting:** $\Gamma = \Gamma_0(p), \chi = \left( \frac{\cdot}{p} \right)$.

**Experimental observation:** $f_+$ and $f_-$ are two different functions with the same eigenvalue.

**Conclusion:** For real even Dirichlet character, in general the spectrum have multiplicity 2.

**Explanation:** If $\varphi$ is invariant under $\chi$ then $\bar{\varphi}$ is invariant under $\bar{\chi} = \chi$, and one can show that in general $\bar{\varphi} \neq \varphi$. 
**Convergence as** $k \to 0$

Setting: $k > 0$, $\Gamma = \text{PSL}(2, \mathbb{Z})$, Maass cusp forms: $\varphi_{j,k}(z)$ and $R = R_j(k)$, $j = 1, 2, \ldots$

**Experimental observation:** As $k \to 0$ the Fourier coefficients of $\varphi_{j,k}(z)$ with $R_j(k) \approx R$ converge to the corresponding coefficients of cusp forms or Eisenstein series with eigenvalue $R$ at weight 0.
Convergence as $k \to 0$

Setting: $k > 0, \Gamma = PSL(2, \mathbb{Z})$, Maass cusp forms: $\phi_{j,k}(z)$ and $R = R_j(k), j = 1, 2, \ldots$.

Experimental observation: As $k \to 0$ the Fourier coefficients of $\phi_{j,k}(z)$ with $R_j(k) \approx R$ converge to the corresponding coefficients of cusp forms or Eisenstein series with eigenvalue $R$ at weight 0.

Explanation: Open question! (Note: This extends results by Hejhal of convergence in “Packets”).
**Convergence as** $k \to 0$

**Experimental observation:** For fixed $j$ we have $R_j(k) \to 0$ and there are no level crossings.
Convergence as $k \rightarrow 0$

Experimental observation: For fixed $j$ we have $R_j(k) \rightarrow 0$ and there are no level crossings.

Conclusion: All weight-zero cusp forms (for $R \leq 20$ or so) are destroyed under perturbation of the weight.
Lift at weight 1

Experimental observation: Fourier coefficients exhibit very strange behaviour: e.g. $a(4) = -a(2) = a(0)$ etc. Some $R$'s are in arithmetic progression.
**Lift at weight 1**

**Experimental observation:** Fourier coefficients exhibit very strange behaviour: e.g. $a(4) = -a(2) = a(0)$ etc. Some $R$’s are in arithmetic progression.

**Explanation:** The eta multiplier at weight 1 is a Dirichlet character on $\Gamma_0(144)$ and by a trivial “lift” we relate the corresponding function spaces. For example if $(12n+1, 13) = 1$ then

$$a(n) = a(0)a(13n+1),$$

and if $(12n-1, 11) = 1$ then

$$a(-n) = \frac{-R^2 a(0)}{a(-1)} a(11n-1).$$
Lift at weight 1

Experimental observation: Fourier coefficients exhibit very strange behaviour: e.g. \( a(4) = -a(2) = a(0) \) etc. Some \( R \)'s are in arithmetic progression.

Explanation: The eta multiplier at weight 1 is a Dirichlet character on \( \Gamma_0(144) \) and by a trivial “lift” we relate the corresponding function spaces. For example if \( (12n + 1, 13) = 1 \) then

\[
a(n) = a(0)a(13n + 1),
\]

and if \( (12n - 1, 11) = 1 \) then

\[
a(-n) = \frac{-R^2a(0)}{a(-1)}a(11n - 1).
\]

Conclusion:
Noncongruence subgroups

Experimental observation: \((\Gamma \text{ is non-congruence, non-cycloidal})\) A plot of the Fourier coefficients, \(c(n)\), in the complex plane:
Noncongruence subgroups

Experimental observation: (Γ is non-congruence, non-cycloidal) A plot of the Fourier coefficients, $c(n)$, in the complex plane:
Noncongruence subgroups

Experimental observation: (\(\Gamma\) is non-congruence, non-cycloidal) A plot of the Fourier coefficients, \(c(n)\), in the complex plane:

![Graph of Fourier coefficients](image)

Explanation: By using symmetries one can show that \(\text{Arg}(c(n))\) depends only on \(n \mod 16\).
**Noncongruence subgroups**

**Experimental observation:** (\(\Gamma\) is non-congruence, non-cycloidal) A plot of the Fourier coefficients, \(c(n)\), in the complex plane:

![Plot of Fourier coefficients](image)

**Explanation:** By using symmetries one can show that \(\text{Arg}(c(n))\) depends only on \(n \mod 16\).

**Open:** How come the standard deviations are different depending on congruence classes?
An illustration of the accuracy

<table>
<thead>
<tr>
<th>Computed:</th>
<th>6.5285026052729949</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formula:</td>
<td>6.528502605272993813463065</td>
</tr>
</tbody>
</table>

Actual difference: $1E-15$

$H(Y_1, Y_2) = 1E-13$

$|c(2)c(3) - c(6)| = 2.6E-15$
# Coefficient Examples

\[ R = 6.52850260527297532 \]

<table>
<thead>
<tr>
<th>( n )</th>
<th>( c(n) ) (computed)</th>
<th>( c(n) ) (formula)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>(-0.0000000000001139)</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>(0.0000000000001231)</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>(1.0000000000002000)</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>(1.0000000000000500)</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>(-0.0000000000000578)</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>(0.0000000000000266)</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>(-0.0000000000000108)</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>(1.0000000000003082)</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>(-0.0000000000000616)</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>(-0.7608214758284901)</td>
<td>(-0.7608214758284897)</td>
</tr>
<tr>
<td>12</td>
<td>(-0.0000000000000964)</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>(-0.0000000000001369)</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>(-1.9705631387316735)</td>
<td>(-1.9705631387317372)</td>
</tr>
<tr>
<td>20</td>
<td>(1.00000000000003788)</td>
<td>1</td>
</tr>
</tbody>
</table>
More examples

\((\Gamma_0(5), \binom{5}{\cdot})\), \(R = 4.89378129143848994\). (Cf. p. 40)

<table>
<thead>
<tr>
<th>(n)</th>
<th>(c^+(n))</th>
<th>(c^-(n))</th>
<th>(\lambda(n))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>-0.48148190237</td>
<td>1.863033149068</td>
<td>1.217161411800i</td>
</tr>
<tr>
<td>3</td>
<td>-0.192808338986</td>
<td>0.451721360518</td>
<td>0.295119713347i</td>
</tr>
<tr>
<td>4</td>
<td>-0.481481902375</td>
<td>-0.481481902377</td>
<td>-0.481481902376</td>
</tr>
<tr>
<td>5</td>
<td>-0.196583115622</td>
<td>1.8034168843792</td>
<td>exp(1.157414657528i)</td>
</tr>
<tr>
<td>6</td>
<td>-0.795198895476</td>
<td>-0.359208326948</td>
<td>-0.3592083269476</td>
</tr>
</tbody>
</table>

\(|\lambda(2)\lambda(3) - \lambda(6)| = 2.6E - 13\).
Formulas for \((\Gamma_0(5), \left(\frac{5}{n}\right))\)

\[
T^*_n = \left(\frac{5}{n}\right) T_n \Rightarrow \lambda(n) = \left(\frac{5}{n}\right) \lambda(n)
\]

\[
\lambda(n) = \begin{cases} 
  i \sqrt{-c^+(n)c^-(n)}, & \left(\frac{5}{n}\right) = -1, \\
  c^+(n) = c^-(n), & \left(\frac{5}{n}\right) = 1,
\end{cases}
\]

and

\[
\lambda(5) = \frac{1}{2} \left( c^+(5) + c^-(5) \right) + \frac{\mu}{2} \left( c^+(5) - c^-(5) \right),
\]

with \(\mu = \frac{\sqrt{c^-(n_0)} - \sqrt{c^+(n_0)}}{\sqrt{c^-(n_0)} + \sqrt{c^+(n_0)}}\), for any \(n_0\) with \(\left(\frac{5}{n_0}\right) = -1\).
Pullback - one cusp
Pullback - Several cusps