Carnegie Mellon University

The Road to Formalising 8-Dimensional Sphere Packing in Lean

Graduate Student and Postdoc Seminar - Department of Mathematics

Sidharth Hariharan

11 September, 2025

sidharthhariharan@cmu.edu

Contents

Formalising Fields Medal-Winning Mathematics

What does it mean to formalise mathematics?

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.

The main advantage is that we can have immense confidence in the correctness of completely formalised proofs.

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.

The main advantage is that we can have immense confidence in the correctness of completely formalised proofs. But there are a few more:

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.

The main advantage is that we can have immense confidence in the correctness of completely formalised proofs. But there are a few more:

1. Standardisation and centralisation

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.

The main advantage is that we can have immense confidence in the correctness of completely formalised proofs. But there are a few more:

1. Standardisation and centralisation
2. Faster review processes for publications

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.

The main advantage is that we can have immense confidence in the correctness of completely formalised proofs. But there are a few more:

1. Standardisation and centralisation
2. Faster review processes for publications
3. Proof discovery

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.

There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.

The main advantage is that we can have immense confidence in the correctness of completely formalised proofs. But there are a few more:

1. Standardisation and centralisation
2. Faster review processes for publications
3. Proof discovery

These applications are to mathematics research. There are more applications still to industry (AI, crypto) and pedagogy.

What does it mean to formalise mathematics?

Typically, formalising a mathematical result means expressing it as computer code whose correctness is checked by a compiler.
There are many computer languages in which mathematics can be expressed, such as Lean, Isabelle, and Agda.
The main advantage is that we can have immense confidence in the correctness of completely formalised proofs. But there are a few more:

1. Standardisation and centralisation
2. Faster review processes for publications
3. Proof discovery

These applications are to mathematics research. There are more applications still to industry (AI, crypto) and pedagogy.

```
▼LPBound.lean:136:2
▼Tactic state
1 goal
d : ℕ
f : 𝓢(EuclideanSpace ℝ (Fin d), ℂ)
hne_zero : f ≠ 0
hReal : ∀ (x : EuclideanSpace ℝ (Fin d)), ↑(f x).re = f x
hRealFourier : ∀ (x : EuclideanSpace ℝ (Fin d)), ↑(𝓕 (⇑f) x).re = 𝓕 (⇑f) x
hCohnElkies₂ : ∀ (x : EuclideanSpace ℝ (Fin d)), (𝓕 (⇑f) x).re ≥ 0
haux₂ : ∫ (v : EuclideanSpace ℝ (Fin d)), 𝓕 (⇑f) v = ↑(∫ (v : EuclideanSpace ℝ (Fin d)), (𝓕 (⇑f) v).re)
haux₁ : (f 0).re = ∫ (v : EuclideanSpace ℝ (Fin d)), (𝓕 (⇑f) v).re
⊢ 0 < (f 0).re
```

Why formalise 8-dimensional sphere packing?

Why formalise 8-dimensional sphere packing?

When Tom Hales formally announced his intent to formalise 3-dimensional sphere packing, his work was still under review. He wrote, "my motivations for this project are far more complex than a simple hope of removing residual doubt from the minds of [a] few referees. Indeed, I see formal methods as fundamental to the long-term growth of mathematics." [3]

Why formalise 8-dimensional sphere packing?

When Tom Hales formally announced his intent to formalise 3-dimensional sphere packing, his work was still under review. He wrote, "my motivations for this project are far more complex than a simple hope of removing residual doubt from the minds of [a] few referees. Indeed, I see formal methods as fundamental to the long-term growth of mathematics." [3]

This project is undertaken in the spirit of those far more complex motivations.

Why formalise 8-dimensional sphere packing?

When Tom Hales formally announced his intent to formalise 3-dimensional sphere packing, his work was still under review. He wrote, "my motivations for this project are far more complex than a simple hope of removing residual doubt from the minds of [a] few referees. Indeed, I see formal methods as fundamental to the long-term growth of mathematics." [3]

This project is undertaken in the spirit of those far more complex motivations. It aims to

Why formalise 8-dimensional sphere packing?

When Tom Hales formally announced his intent to formalise 3-dimensional sphere packing, his work was still under review. He wrote, "my motivations for this project are far more complex than a simple hope of removing residual doubt from the minds of [a] few referees. Indeed, I see formal methods as fundamental to the long-term growth of mathematics." [3]

This project is undertaken in the spirit of those far more complex motivations. It aims to

1. solidify the credibility not only of this result but of those that build on it;

Why formalise 8-dimensional sphere packing?

When Tom Hales formally announced his intent to formalise 3-dimensional sphere packing, his work was still under review. He wrote, “my motivations for this project are far more complex than a simple hope of removing residual doubt from the minds of [a] few referees. Indeed, I see formal methods as fundamental to the long-term growth of mathematics.” [3]

This project is undertaken in the spirit of those far more complex motivations. It aims to

1. solidify the credibility not only of this result but of those that build on it;
2. test the capabilities of Lean and `mathlib` against modern, groundbreaking mathematics;

Why formalise 8-dimensional sphere packing?

When Tom Hales formally announced his intent to formalise 3-dimensional sphere packing, his work was still under review. He wrote, “my motivations for this project are far more complex than a simple hope of removing residual doubt from the minds of [a] few referees. Indeed, I see formal methods as fundamental to the long-term growth of mathematics.” [3]

This project is undertaken in the spirit of those far more complex motivations. It aims to

1. solidify the credibility not only of this result but of those that build on it;
2. test the capabilities of Lean and `mathlib` against modern, groundbreaking mathematics; and
3. expand existing code and methodologies for tackling computational results in analysis.

Why formalise 8-dimensional sphere packing?

When Tom Hales formally announced his intent to formalise 3-dimensional sphere packing, his work was still under review. He wrote, "my motivations for this project are far more complex than a simple hope of removing residual doubt from the minds of [a] few referees. Indeed, I see formal methods as fundamental to the long-term growth of mathematics." [3]

This project is undertaken in the spirit of those far more complex motivations. It aims to

1. solidify the credibility not only of this result but of those that build on it;
2. test the capabilities of Lean and `mathlib` against modern, groundbreaking mathematics; and
3. expand existing code and methodologies for tackling computational results in analysis.

Formalising such a recent Fields Medal-winning result would be a monumental breakthrough for the formal theorem proving community.

Who’s working on this project, and how’s it going?

Who’s working on this project, and how’s it going?

Formalising Sphere Packing in Lean

by Chris Birkbeck, Sidharth Hariharan, Gareth Ma, Bhavik Mehta, Seewoo Lee, Maryna Viazovska

Blueprint (web) | Blueprint (pdf) | Documentation | GitHub

Formalising Sphere Packing in Lean

In 2022, Maryna Viazovska was awarded the Fields Medal for solving the sphere packing problem in dimension 8. This project formalises this result in the Lean Theorem Prover following her original paper and followup calculations by Lee. It is an online, open-source collaboration currently being led by Christopher Birkbeck, Sidharth Hariharan, Bhavik Mehta and Seewoo Lee. If you’d like to contribute, you may find the following links useful!

- Zulip chat for coordination
- Blueprint
- Blueprint as pdf
- Dependency graph
- Doc pages for this repository
- Project dashboard
- Viazovska’s original paper

SCAN TO VISIT OUR WEBSITE!

Who’s working on this project, and how’s it going?

Formalising Sphere Packing in Lean

by Chris Birkbeck, Sidharth Hariharan, Gareth Ma, Bhavik Mehta, Seewoo Lee, Maryna Viazovska

Blueprint (web) | Blueprint (pdf) | Documentation | GitHub

Formalising Sphere Packing in Lean

In 2022, Maryna Viazovska was awarded the Fields Medal for solving the sphere packing problem in dimension 8. This project formalises this result in the Lean Theorem Prover following her original paper and followup calculations by Lee. It is an online, open-source collaboration currently being led by Christopher Birkbeck, Sidharth Hariharan, Bhavik Mehta and Seewoo Lee. If you’d like to contribute, you may find the following links useful!

- Zulip chat for coordination
- Blueprint
- Blueprint as pdf
- Dependency graph
- Doc pages for this repository
- Project dashboard
- Viazovska’s original paper

SCAN TO VISIT OUR WEBSITE!

1. March 2024:

Who’s working on this project, and how’s it going?

Formalising Sphere Packing in Lean

by Chris Birkbeck, Sidharth Hariharan, Gareth Ma, Bhavik Mehta, Seewoo Lee, Maryna Viazovska

Blueprint (web) | Blueprint (pdf) | Documentation | GitHub

Formalising Sphere Packing in Lean

In 2022, Maryna Viazovska was awarded the Fields Medal for solving the sphere packing problem in dimension 8. This project formalises this result in the Lean Theorem Prover following her original paper and followup calculations by Lee. It is an online, open-source collaboration currently being led by Christopher Birkbeck, Sidharth Hariharan, Bhavik Mehta and Seewoo Lee. If you’d like to contribute, you may find the following links useful!

- Zulip chat for coordination
- Blueprint
- Blueprint as pdf
- Dependency graph
- Doc pages for this repository
- Project dashboard
- Viazovska’s original paper

SCAN TO VISIT OUR WEBSITE!

1. March 2024: Viazovska and I kickstart project in Lausanne.

Who’s working on this project, and how’s it going?

Formalising Sphere Packing in Lean

by Chris Birkbeck, Sidharth Hariharan, Gareth Ma, Bhavik Mehta, Seewoo Lee, Maryna Viazovska

Blueprint (web) | Blueprint (pdf) | Documentation | GitHub

Formalising Sphere Packing in Lean

In 2022, Maryna Viazovska was awarded the Fields Medal for solving the sphere packing problem in dimension 8. This project formalises this result in the Lean Theorem Prover following her original paper and followup calculations by Lee. It is an online, open-source collaboration currently being led by Christopher Birkbeck, Sidharth Hariharan, Bhavik Mehta and Seewoo Lee. If you’d like to contribute, you may find the following links useful!

- Zulip chat for coordination
- Blueprint
- Blueprint as pdf
- Dependency graph
- Doc pages for this repository
- Project dashboard
- Viazovska’s original paper

SCAN TO VISIT OUR WEBSITE!

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:

Who’s working on this project, and how’s it going?

Formalising Sphere Packing in Lean

by Chris Birkbeck, Sidharth Hariharan, Gareth Ma, Bhavik Mehta, Seewoo Lee, Maryna Viazovska

Blueprint (web) | Blueprint (pdf) | Documentation | GitHub

Formalising Sphere Packing in Lean

In 2022, Maryna Viazovska was awarded the Fields Medal for solving the sphere packing problem in dimension 8. This project formalises this result in the Lean Theorem Prover following her original paper and followup calculations by Lee. It is an online, open-source collaboration currently being led by Christopher Birkbeck, Sidharth Hariharan, Bhavik Mehta and Seewoo Lee. If you’d like to contribute, you may find the following links useful!

- Zulip chat for coordination
- Blueprint
- Blueprint as pdf
- Dependency graph
- Doc pages for this repository
- Project dashboard
- Viazovska’s original paper

SCAN TO VISIT OUR WEBSITE!

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.

Who's working on this project, and how's it going?

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.
   - I formalise much of an important dependency courtesy Cohn-Elkies [1, Theorem 3.1].

Who’s working on this project, and how’s it going?

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.
   - I formalise much of an important dependency courtesy Cohn-Elkies [1, Theorem 3.1].
   - Birkbeck and Lee begin formalising facts about modular forms.

Who’s working on this project, and how’s it going?

Formalising Sphere Packing in Lean

by Chris Birkbeck, Sidharth Hariharan, Gareth Ma, Bhavik Mehta, Seewoo Lee, Maryna Viazovska

Blueprint (web) | Blueprint (pdf) | Documentation | GitHub

Formalising Sphere Packing in Lean

In 2022, Maryna Viazovska was awarded the Fields Medal for solving the sphere packing problem in dimension 8. This project formalises this result in the Lean Theorem Prover following her original paper and followup calculations by Lee. It is an online, open-source collaboration currently being led by Christopher Birkbeck, Sidharth Hariharan, Bhavik Mehta and Seewoo Lee. If you’d like to contribute, you may find the following links useful!

- Zulip chat for coordination
- Blueprint
- Blueprint as pdf
- Dependency graph
- Doc pages for this repository
- Project dashboard
- Viazovska’s original paper

SCAN TO VISIT OUR WEBSITE!

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.
   - I formalise much of an important dependency courtesy Cohn-Elkies [1, Theorem 3.1].
   - Birkbeck and Lee begin formalising facts about modular forms.
3. October 2024-June 2025:

Who's working on this project, and how's it going?

Formalising Sphere Packing in Lean

by Chris Birkbeck, Sidharth Hariharan, Gareth Ma, Bhavik Mehta, Seewoo Lee, Maryna Viazovska

Blueprint (web) | Blueprint (pdf) | Documentation | GitHub

Formalising Sphere Packing in Lean

In 2022, Maryna Viazovska was awarded the Fields Medal for solving the sphere packing problem in dimension 8. This project formalises this result in the Lean Theorem Prover following her original paper and followup calculations by Lee. It is an online, open-source collaboration currently being led by Christopher Birkbeck, Sidharth Hariharan, Bhavik Mehta and Seewoo Lee. If you'd like to contribute, you may find the following links useful!

- Zulip chat for coordination
- Blueprint
- Blueprint as pdf
- Dependency graph
- Doc pages for this repository
- Project dashboard
- Viazovska's original paper

SCAN TO VISIT OUR WEBSITE!

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.
   - I formalise much of an important dependency courtesy Cohn-Elkies [1, Theorem 3.1].
   - Birkbeck and Lee begin formalising facts about modular forms.
3. October 2024-June 2025:
   - I work on formalising Viazovska's magic function for my Master's project at Imperial under Bhavik Mehta.

Who's working on this project, and how's it going?

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.
   - I formalise much of an important dependency courtesy Cohn-Elkies [1, Theorem 3.1].
   - Birkbeck and Lee begin formalising facts about modular forms.
3. October 2024-June 2025:
   - I work on formalising Viazovska's magic function for my Master's project at Imperial under Bhavik Mehta.
   - Birkbeck and Lee continue formalising results on (quasi)modular forms.

Who’s working on this project, and how’s it going?

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.
   - I formalise much of an important dependency courtesy Cohn-Elkies [1, Theorem 3.1].
   - Birkbeck and Lee begin formalising facts about modular forms.
3. October 2024-June 2025:
   - I work on formalising Viazovska’s magic function for my Master’s project at Imperial under Bhavik Mehta.
   - Birkbeck and Lee continue formalising results on (quasi)modular forms.
4. 13 June, 2025:

Who's working on this project, and how's it going?

1. March 2024: Viazovska and I kickstart project in Lausanne.
2. July-September 2024:
   - Ma and I formalise basic theory.
   - I formalise much of an important dependency courtesy Cohn-Elkies [1, Theorem 3.1].
   - Birkbeck and Lee begin formalising facts about modular forms.
3. October 2024-June 2025:
   - I work on formalising Viazovska's magic function for my Master's project at Imperial under Bhavik Mehta.
   - Birkbeck and Lee continue formalising results on (quasi)modular forms.
4. 13 June, 2025: Viazovska unveils project to public at Big Proof 2025, Cambridge.

Viazovska’s Magical Construction

The problem, the solution, and the way to get there

The sphere packing problem asks the following question:

The problem, the solution, and the way to get there

The sphere packing problem asks the following question:

What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

The problem, the solution, and the way to get there

The sphere packing problem asks the following question:

> What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4].

The sphere packing problem asks the following question:

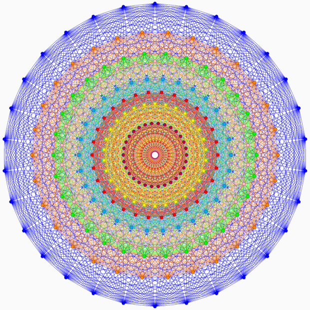
> What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4]. Below, we give an illustration of a projection of the E_8 root system onto its so-called Coxeter plane [2].

The sphere packing problem asks the following question:

> What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

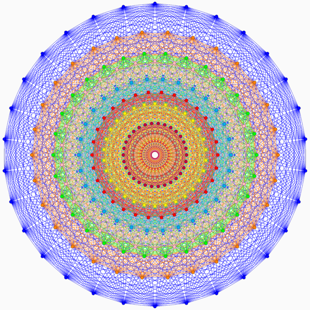
Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4]. Below, we give an illustration of a projection of the E_8 root system onto its so-called Coxeter plane [2].



The sphere packing problem asks the following question:

> What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4]. Below, we give an illustration of a projection of the E_8 root system onto its so-called Coxeter plane [2]. Nodes are vectors and edges connect nearest neighbours.



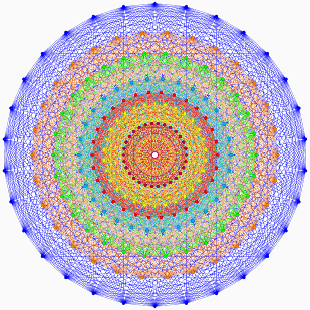
The problem, the solution, and the way to get there

The sphere packing problem asks the following question:

What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4]. Below, we give an illustration of a projection of the E_8 root system onto its so-called Coxeter plane [2]. Nodes are vectors and edges connect nearest neighbours.

This was long suspected, and in 2003, Henry Cohn and Noam Elkies proposed a method that many believed would offer a means to a proof.



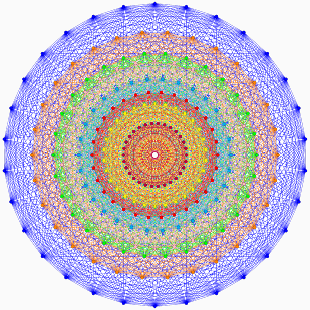
The problem, the solution, and the way to get there

The sphere packing problem asks the following question:

> What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4]. Below, we give an illustration of a projection of the E_8 root system onto its so-called Coxeter plane [2]. Nodes are vectors and edges connect nearest neighbours.

This was long suspected, and in 2003, Henry Cohn and Noam Elkies proposed a method that many believed would offer a means to a proof. They showed that if a certain type of $\mathbb{R}^n \to \mathbb{R}$ function is constructed, then all sphere packings in $\mathbb{R}^n$ are bounded above by a certain quantity that depends on the function.

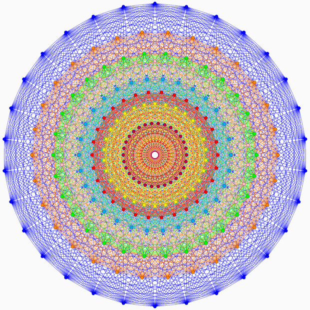


The sphere packing problem asks the following question:

What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4]. Below, we give an illustration of a projection of the E_8 root system onto its so-called Coxeter plane [2]. Nodes are vectors and edges connect nearest neighbours.

This was long suspected, and in 2003, Henry Cohn and Noam Elkies proposed a method that many believed would offer a means to a proof. They showed that if a certain type of $\mathbb{R}^n \to \mathbb{R}$ function is constructed, then all sphere packings in $\mathbb{R}^n$ are bounded above by a certain quantity that depends on the function.



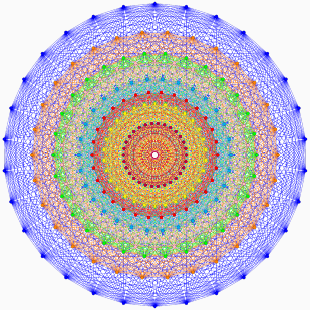
Unfortunately, constructing such functions is extremely difficult.

The sphere packing problem asks the following question:

What is the densest arrangement in $\mathbb{R}^n$ of non-overlapping n-spheres of the same radius?

Viazovska famously showed that the densest arrangement in $\mathbb{R}^8$ corresponds to spheres centred at points on the E_8 root lattice [4]. Below, we give an illustration of a projection of the E_8 root system onto its so-called Coxeter plane [2]. Nodes are vectors and edges connect nearest neighbours.

This was long suspected, and in 2003, Henry Cohn and Noam Elkies proposed a method that many believed would offer a means to a proof. They showed that if a certain type of $\mathbb{R}^n \to \mathbb{R}$ function is constructed, then all sphere packings in $\mathbb{R}^n$ are bounded above by a certain quantity that depends on the function.



Unfortunately, constructing such functions is extremely difficult. Viazovska's monumental breakthrough was to construct such a function using the theory of modular forms.

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in $\mathbb{R}^d$**.

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24.

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

A Word on Schwartz Functions

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

A Word on Schwartz Functions

Schwartz functions are smooth and rapidly decaying functions.

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

A Word on Schwartz Functions

Schwartz functions are smooth and rapidly decaying functions. More precisely, $f : \mathbb{R}^d \to \mathbb{R}$ is Schwartz if $\forall k, n \in \mathbb{N}$, $\exists C \in \mathbb{R}$ such that

$$\left\|f^{(n)}(x)\right\| \cdot \|x\|^k \leq C$$

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

A Word on Schwartz Functions

Schwartz functions are smooth and rapidly decaying functions. More precisely, $f : \mathbb{R}^d \to \mathbb{R}$ is Schwartz if $\forall k, n \in \mathbb{N}, \exists C \in \mathbb{R}$ such that

$$\left\|f^{(n)}(x)\right\| \cdot \|x\|^k \leq C$$

1. The set of $\mathbb{R}^d \to \mathbb{R}$ Schwartz functions is an $\mathbb{R}$-vector space, called the Schwartz space.

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathbf{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

A Word on Schwartz Functions

Schwartz functions are smooth and rapidly decaying functions. More precisely, $f : \mathbb{R}^d \to \mathbb{R}$ is Schwartz if $\forall k, n \in \mathbb{N}$, $\exists C \in \mathbb{R}$ such that

$$\left\| f^{(n)}(x) \right\| \cdot \|x\|^k \leq C$$

1. The set of $\mathbb{R}^d \to \mathbb{R}$ Schwartz functions is an $\mathbb{R}$-vector space, called the Schwartz space.
2. The Fourier transform $\mathcal{F}$ is a linear isomorphism of this space.

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

A Word on Schwartz Functions

Schwartz functions are smooth and rapidly decaying functions. More precisely, $f : \mathbb{R}^d \to \mathbb{R}$ is Schwartz if $\forall k, n \in \mathbb{N}$, $\exists C \in \mathbb{R}$ such that

$$\left\| f^{(n)}(x) \right\| \cdot \|x\|^k \leq C$$

1. The set of $\mathbb{R}^d \to \mathbb{R}$ Schwartz functions is an $\mathbb{R}$-vector space, called the Schwartz space.
2. The Fourier transform $\mathcal{F}$ is a linear isomorphism of this space.
3. Composition with any function satisfying some growth conditions is a linear map of Schwartz spaces.

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Dependencies of Cohn-Elkies

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Dependencies of Cohn-Elkies

1. Enough to consider periodic packings:

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Dependencies of Cohn-Elkies

1. Enough to consider periodic packings: Cohn-Elkies showed [1, Appendix A] that if the bound holds for all periodic packings, it holds for all packings.

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Dependencies of Cohn-Elkies

1. Enough to consider periodic packings:
   Cohn-Elkies showed [1, Appendix A] that if the bound holds for all periodic packings, it holds for all packings.
   This is being worked on by Yaël Dillies.

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Dependencies of Cohn-Elkies

1. Enough to consider periodic packings:
   Cohn-Elkies showed [1, Appendix A] that if the bound holds for all periodic packings, it holds for all packings.
   This is being worked on by Yaël Dillies.
2. The Poisson Summation Formula in $\mathbb{R}^n$:

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Dependencies of Cohn-Elkies

1. <u>Enough to consider periodic packings:</u>
   Cohn-Elkies showed [1, Appendix A] that if the bound holds for all periodic packings, it holds for all packings.
   This is being worked on by Yaël Dillies.
2. <u>The Poisson Summation Formula in $\mathbb{R}^n$:</u>
   Let let $\Lambda \subset \mathbb{R}^d$ be a lattice. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a Schwartz function. For all $v \in \mathbb{R}^d$,

$$\sum_{\ell \in \Lambda} f(\ell + v) = \frac{1}{\mathrm{Vol}\left(\mathbb{R}^d / \Lambda\right)} \sum_{m \in \Lambda^*} \widehat{f}(y) e^{-2\pi i \langle v, m \rangle}$$

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Dependencies of Cohn-Elkies

1. Enough to consider periodic packings:
   Cohn-Elkies showed [1, Appendix A] that if the bound holds for all periodic packings, it holds for all packings.
   This is being worked on by Yaël Dillies.
2. The Poisson Summation Formula in $\mathbb{R}^n$:
   Let let $\Lambda \subset \mathbb{R}^d$ be a lattice. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a Schwartz function. For all $v \in \mathbb{R}^d$,

$$\sum_{\ell \in \Lambda} f(\ell + v) = \frac{1}{\mathrm{Vol}\left(\mathbb{R}^d / \Lambda\right)} \sum_{m \in \Lambda^*} \widehat{f}(y) e^{-2\pi i \langle v, m \rangle}$$

   *I am working on this, and **I need help!** If you're interested, **please reach out!***

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

1. <u>Restrict to radial Schwartz functions:</u>

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

1. <u>Restrict to radial Schwartz functions:</u>
   - Allows us to consider $\mathbb{R} \to \mathbb{R}$ Schwartz functions - "Schwartzness bridge"

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

1. <u>Restrict to radial Schwartz functions:</u>
   - Allows us to consider $\mathbb{R} \to \mathbb{R}$ Schwartz functions - "Schwartzness bridge"
   - If f is radial and Schwartz, then there are unique Fourier ± 1-eigenfunctions

$$f_+ = \frac{f + \widehat{f}}{2} \qquad f_- = \frac{f - \widehat{f}}{2}$$

such that $f = f_+ + f_-$ and $\mathcal{F}(f_\pm) = \pm f_\pm$.

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

1. Restrict to radial Schwartz functions:
    - Allows us to consider $\mathbb{R} \to \mathbb{R}$ Schwartz functions - "Schwartzness bridge"
    - If f is radial and Schwartz, then there are unique Fourier ± 1-eigenfunctions

    $$f_+ = \frac{f + \widehat{f}}{2} \qquad f_- = \frac{f - \widehat{f}}{2}$$

    such that $f = f_+ + f_-$ and $\mathcal{F}(f_\pm) = \pm f_\pm$.
2. Double zeroes at (most) lattice points:

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

1. Restrict to radial Schwartz functions:
   - Allows us to consider $\mathbb{R} \to \mathbb{R}$ Schwartz functions - "Schwartzness bridge"
   - If f is radial and Schwartz, then there are unique Fourier ± 1-eigenfunctions

   $$f_+ = \frac{f + \widehat{f}}{2} \qquad f_- = \frac{f - \widehat{f}}{2}$$

   such that $f = f_+ + f_-$ and $\mathcal{F}(f_\pm) = \pm f_\pm$.
2. Double zeroes at (most) lattice points:
   - Single zeroes can be seen by inspecting proof of Cohn-Elkies.

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

1. Restrict to radial Schwartz functions:
   - Allows us to consider $\mathbb{R} \to \mathbb{R}$ Schwartz functions - "Schwartzness bridge"
   - If f is radial and Schwartz, then there are unique Fourier ± 1-eigenfunctions

   $$f_+ = \frac{f + \widehat{f}}{2} \qquad f_- = \frac{f - \widehat{f}}{2}$$

   such that $f = f_+ + f_-$ and $\mathcal{F}(f_\pm) = \pm f_\pm$.
2. Double zeroes at (most) lattice points:
   - Single zeroes can be seen by inspecting proof of Cohn-Elkies.
   - Double zeroes can be seen by inspecting the conditions of Cohn-Elkies.

From geometry to analysis: Cohn and Elkies's upper bounds for sphere packing

In 2003, Cohn and Elkies proved [1, Theorem 3.1] that if $f : \mathbb{R}^d \to \mathbb{R}$ is a Schwartz function and

1. f is not identically zero;
2. for all $x \in \mathbb{R}^d$, if $\|x\| \geq 1$ then $f(x) \leq 0$; and
3. for all $x \in \mathbb{R}^d$, $\widehat{f}(x) \geq 0$,

then the densities of all sphere packings in $\mathbb{R}^n$ are bounded above by

$$\frac{f(0)}{\widehat{f}(0)} \cdot \mathrm{Vol}\left(B_d\left(0, \frac{1}{2}\right)\right)$$

If f satisfies the above and its upper-bound is optimal, we say f is **magic in** $\mathbb{R}^d$. Thus far, magic functions have only been found in dimensions 8 and 24. It is believed that they also exist in dimension 2, but nowhere else.

Insights from Cohn-Elkies

1. Restrict to radial Schwartz functions:
   - Allows us to consider $\mathbb{R} \to \mathbb{R}$ Schwartz functions - "Schwartzness bridge"
   - If f is radial and Schwartz, then there are unique Fourier ± 1-eigenfunctions

     $$f_+ = \frac{f + \widehat{f}}{2} \qquad f_- = \frac{f - \widehat{f}}{2}$$

     such that $f = f_+ + f_-$ and $\mathcal{F}(f_\pm) = \pm f_\pm$.
2. Double zeroes at (most) lattice points:
   - Single zeroes can be seen by inspecting proof of Cohn-Elkies.
   - Double zeroes can be seen by inspecting the conditions of Cohn-Elkies.

We need **Fourier eigenfunctions with double zeroes at lattice points**.

Viazovska's Fourier eigenfunctions

Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b.

Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b. She then simply scaled it to obtain a magic function f whose corresponding bound was the density of the E_8 packing.

Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b. She then simply scaled it to obtain a magic function f whose corresponding bound was the density of the E_8 packing. a, b, g and f are defined as follows.

Viazovska's Fourier eigenfunctions

Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b. She then simply scaled it to obtain a magic function f whose corresponding bound was the density of the E_8 packing. a, b, g and f are defined as follows.

$$a(x) := \int_{-1}^{i} \phi_0\left(\frac{-1}{z+1}\right) (z+1)^2 \; e^{\pi i \|x\|^2 z} \, dz + \int_{1}^{i} \phi_0\left(\frac{-1}{z-1}\right) (z-1)^2 \; e^{\pi i \|x\|^2 z} \, dz \\ - 2\int_{0}^{i} \phi_0\left(\frac{-1}{z}\right) z^2 \, e^{\pi i \|x\|^2 z} \, dz + 2\int_{i}^{i\infty} \phi_0(z) \; e^{\pi i \|x\|^2 z} \, dz$$

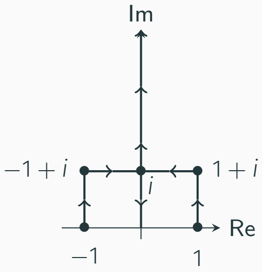
Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b. She then simply scaled it to obtain a magic function f whose corresponding bound was the density of the E_8 packing. a, b, g and f are defined as follows.

$$\begin{aligned} a(x) := &\int_{-1}^{i} \phi_0\left(\frac{-1}{z+1}\right) (z+1)^2 \; e^{\pi i \|x\|^2 z} \, dz + \int_{1}^{i} \phi_0\left(\frac{-1}{z-1}\right) (z-1)^2 \; e^{\pi i \|x\|^2 z} \, dz \\ &- 2\int_{0}^{i} \phi_0\left(\frac{-1}{z}\right) z^2 \, e^{\pi i \|x\|^2 z} \, dz + 2\int_{i}^{i\infty} \phi_0(z) \; e^{\pi i \|x\|^2 z} \, dz \\ b(x) := &\int_{-1}^{i} \psi_S\left(\frac{-1}{z+1}\right) (z+1)^2 \; e^{\pi i \|x\|^2 z} \, dz + \int_{1}^{i} \psi_S\left(\frac{-1}{z-1}\right) (z-1)^2 \; e^{\pi i \|x\|^2 z} \, dz \\ &- 2\int_{0}^{i} \psi_S\left(\frac{-1}{z}\right) z^2 \, e^{\pi i \|x\|^2 z} \, dz - 2\int_{i}^{i\infty} \psi_S(z) \; e^{\pi i \|x\|^2 z} \, dz \end{aligned}$$

Viazovska's Fourier eigenfunctions

Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b. She then simply scaled it to obtain a magic function f whose corresponding bound was the density of the E_8 packing. a, b, g and f are defined as follows.

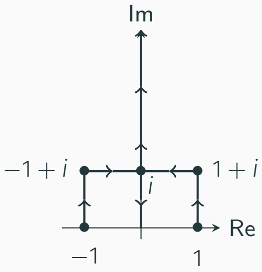
$$\begin{aligned} a(x) := &\int_{-1}^{i} \phi_0\left(\frac{-1}{z+1}\right) (z+1)^2 \; e^{\pi i \|x\|^2 z}\, dz + \int_{1}^{i} \phi_0\left(\frac{-1}{z-1}\right) (z-1)^2 \; e^{\pi i \|x\|^2 z}\, dz \\ &- 2\int_{0}^{i} \phi_0\left(\frac{-1}{z}\right) z^2\, e^{\pi i \|x\|^2 z}\, dz + 2\int_{i}^{i\infty} \phi_0(z)\; e^{\pi i \|x\|^2 z}\, dz \\ b(x) := &\int_{-1}^{i} \psi_S\left(\frac{-1}{z+1}\right) (z+1)^2 \; e^{\pi i \|x\|^2 z}\, dz + \int_{1}^{i} \psi_S\left(\frac{-1}{z-1}\right) (z-1)^2 \; e^{\pi i \|x\|^2 z}\, dz \\ &- 2\int_{0}^{i} \psi_S\left(\frac{-1}{z}\right) z^2\, e^{\pi i \|x\|^2 z}\, dz - 2\int_{i}^{i\infty} \psi_S(z)\; e^{\pi i \|x\|^2 z}\, dz \end{aligned}$$



Viazovska's Fourier eigenfunctions

Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b. She then simply scaled it to obtain a magic function f whose corresponding bound was the density of the E_8 packing. a, b, g and f are defined as follows.

$$\begin{aligned} a(x) :=& \int_{-1}^{i} \phi_0\left(\frac{-1}{z+1}\right)(z+1)^2\, e^{\pi i \|x\|^2 z}\, dz + \int_{1}^{i} \phi_0\left(\frac{-1}{z-1}\right)(z-1)^2\, e^{\pi i \|x\|^2 z}\, dz \\ &- 2\int_{0}^{i} \phi_0\left(\frac{-1}{z}\right) z^2\, e^{\pi i \|x\|^2 z}\, dz + 2\int_{i}^{i\infty} \phi_0(z)\, e^{\pi i \|x\|^2 z}\, dz \\ b(x) :=& \int_{-1}^{i} \psi_S\left(\frac{-1}{z+1}\right)(z+1)^2\, e^{\pi i \|x\|^2 z}\, dz + \int_{1}^{i} \psi_S\left(\frac{-1}{z-1}\right)(z-1)^2\, e^{\pi i \|x\|^2 z}\, dz \\ &- 2\int_{0}^{i} \psi_S\left(\frac{-1}{z}\right) z^2\, e^{\pi i \|x\|^2 z}\, dz - 2\int_{i}^{i\infty} \psi_S(z)\, e^{\pi i \|x\|^2 z}\, dz \\ g(x) :=& \frac{\pi i}{8640} a(x) + \frac{i}{240\pi} b(x) \end{aligned}$$

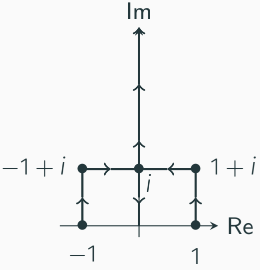


Viazovska constructed a radial Schwartz function g as a linear combination of a Fourier $+1$-eigenfunction a and a -1-eigenfunction b. She then simply scaled it to obtain a magic function f whose corresponding bound was the density of the E_8 packing. a, b, g and f are defined as follows.

$$a(x) := \int_{-1}^{i} \phi_0\left(\frac{-1}{z+1}\right)(z+1)^2\, e^{\pi i \|x\|^2 z}\, dz + \int_{1}^{i} \phi_0\left(\frac{-1}{z-1}\right)(z-1)^2\, e^{\pi i \|x\|^2 z}\, dz$$
$$- 2\int_{0}^{i} \phi_0\left(\frac{-1}{z}\right) z^2\, e^{\pi i \|x\|^2 z}\, dz + 2\int_{i}^{i\infty} \phi_0(z)\, e^{\pi i \|x\|^2 z}\, dz$$

$$b(x) := \int_{-1}^{i} \psi_S\left(\frac{-1}{z+1}\right)(z+1)^2\, e^{\pi i \|x\|^2 z}\, dz + \int_{1}^{i} \psi_S\left(\frac{-1}{z-1}\right)(z-1)^2\, e^{\pi i \|x\|^2 z}\, dz$$
$$- 2\int_{0}^{i} \psi_S\left(\frac{-1}{z}\right) z^2\, e^{\pi i \|x\|^2 z}\, dz - 2\int_{i}^{i\infty} \psi_S(z)\, e^{\pi i \|x\|^2 z}\, dz$$

$$g(x) := \frac{\pi i}{8640} a(x) + \frac{i}{240\pi} b(x) \qquad f(x) := g\left(\sqrt{2}x\right)$$



ϕ_0 and ψ_S: the role of modular forms

ϕ_0 and ψ_S: the role of modular forms

Modular forms are holomorphic functions from the upper half-plane $\mathbb{H}$ to $\mathbb{C}$ that satisfy (weighted) invariance properties under Möbius integer changes of variables.

Modular forms are holomorphic functions from the upper half-plane $\mathbb{H}$ to $\mathbb{C}$ that satisfy (weighted) invariance properties under Möbius integer changes of variables. For example, a "**weight 4, level 1**" modular form $\alpha : \mathbb{H} \to \mathbb{C}$ would satisfy

$$\alpha(z) = (cz+d)^{-4}\ \alpha\left(\frac{az+b}{cz+d}\right) \qquad \text{for all } z \in \mathbb{H} \text{ and } \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}(2,\mathbb{Z})$$

Modular forms are holomorphic functions from the upper half-plane $\mathbb{H}$ to $\mathbb{C}$ that satisfy (weighted) invariance properties under Möbius integer changes of variables. For example, a "**weight 4, level 1**" modular form $\alpha : \mathbb{H} \to \mathbb{C}$ would satisfy

$$\alpha(z) = (cz+d)^{-4}\ \alpha\left(\frac{az+b}{cz+d}\right) \qquad \text{for all } z \in \mathbb{H} \text{ and } \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}(2,\mathbb{Z})$$

The definitions of the integrands making up a and b in terms of modular forms endow them with important change-of-variable symmetries.

ϕ_0 and ψ_S: the role of modular forms

Modular forms are holomorphic functions from the upper half-plane $\mathbb{H}$ to $\mathbb{C}$ that satisfy (weighted) invariance properties under Möbius integer changes of variables. For example, a "**weight 4, level 1**" modular form $\alpha : \mathbb{H} \to \mathbb{C}$ would satisfy

$$\alpha(z) = (cz+d)^{-4}\ \alpha\left(\frac{az+b}{cz+d}\right) \qquad \text{for all } z \in \mathbb{H} \text{ and } \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}(2,\mathbb{Z})$$

The definitions of the integrands making up a and b in terms of modular forms endow them with important change-of-variable symmetries. For instance,

$$\psi_S\left(\frac{-1}{z-1}\right)\ (z-1)^2 = \psi_S\left(\frac{-1}{z+1}\right)\ (z+1)^2$$

for all $z \in \mathbb{H}$.

ϕ_0 and ψ_S: the role of modular forms

Modular forms are holomorphic functions from the upper half-plane $\mathbb{H}$ to $\mathbb{C}$ that satisfy (weighted) invariance properties under Möbius integer changes of variables. For example, a "**weight 4, level 1**" modular form $\alpha : \mathbb{H} \to \mathbb{C}$ would satisfy

$$\alpha(z) = (cz+d)^{-4}\ \alpha\left(\frac{az+b}{cz+d}\right) \qquad \text{for all } z \in \mathbb{H} \text{ and } \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}(2,\mathbb{Z})$$

The definitions of the integrands making up a and b in terms of modular forms endow them with important change-of-variable symmetries. For instance,

$$\psi_S\left(\frac{-1}{z-1}\right)\ (z-1)^2 = \psi_S\left(\frac{-1}{z+1}\right)\ (z+1)^2$$

for all $z \in \mathbb{H}$. Denote this by $\psi_T(z)$.

ϕ_0 and ψ_S: the role of modular forms

Modular forms are holomorphic functions from the upper half-plane $\mathbb{H}$ to $\mathbb{C}$ that satisfy (weighted) invariance properties under Möbius integer changes of variables. For example, a "**weight 4, level 1**" modular form $\alpha : \mathbb{H} \to \mathbb{C}$ would satisfy

$$\alpha(z) = (cz+d)^{-4}\ \alpha\left(\frac{az+b}{cz+d}\right) \qquad \text{for all } z \in \mathbb{H} \text{ and } \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}(2, \mathbb{Z})$$

The definitions of the integrands making up a and b in terms of modular forms endow them with important change-of-variable symmetries. For instance,

$$\psi_S\left(\frac{-1}{z-1}\right)\ (z-1)^2 = \psi_S\left(\frac{-1}{z+1}\right)\ (z+1)^2$$

for all $z \in \mathbb{H}$. Denote this by $\psi_T(z)$. We further have that

$$\psi_T(z) = z^2\ \psi_T\left(\frac{-1}{z}\right)$$

ϕ_0 and ψ_S: the role of modular forms

Modular forms are holomorphic functions from the upper half-plane $\mathbb{H}$ to $\mathbb{C}$ that satisfy (weighted) invariance properties under Möbius integer changes of variables. For example, a "**weight 4, level 1**" modular form $\alpha : \mathbb{H} \to \mathbb{C}$ would satisfy

$$\alpha(z) = (cz+d)^{-4}\ \alpha\left(\frac{az+b}{cz+d}\right) \qquad \text{for all } z \in \mathbb{H} \text{ and } \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}(2,\mathbb{Z})$$

The definitions of the integrands making up a and b in terms of modular forms endow them with important change-of-variable symmetries. For instance,

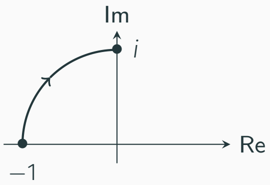
$$\psi_S\left(\frac{-1}{z-1}\right)\ (z-1)^2 = \psi_S\left(\frac{-1}{z+1}\right)\ (z+1)^2$$

for all $z \in \mathbb{H}$. Denote this by $\psi_T(z)$. We further have that

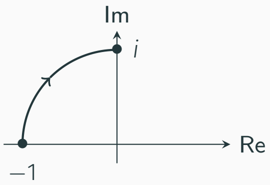
$$\psi_T(z) = z^2\,\psi_T\left(\frac{-1}{z}\right)$$

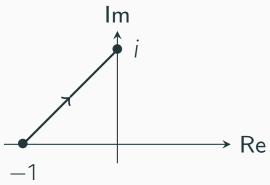
This is crucial to establish that $\widehat{b} = -b$.

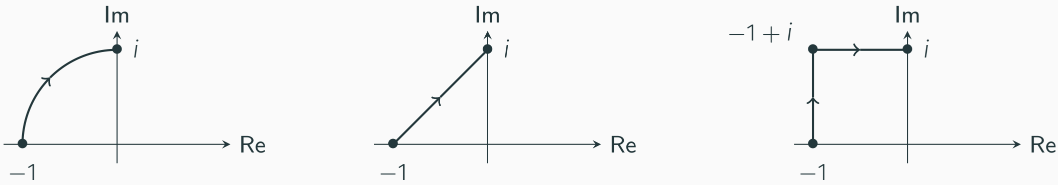
Contour Integration, Informally and Formally


Im
i
Re
-1

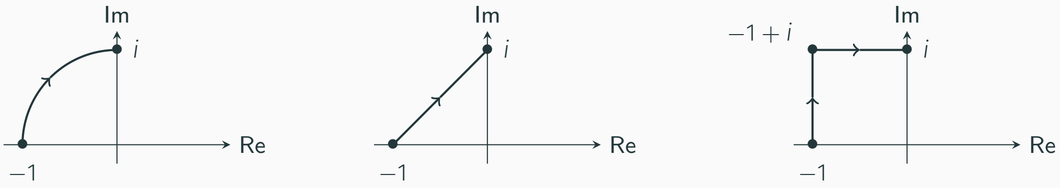
Contour integration






Im
i
Re
-1
Im
i
Re
-1
$-1+i$
Im
i
Re
-1

Contour integration




```
theorem Complex.integral_boundary_rect_eq_zero_of_differentiable_on_off_countable        source
    {E : Type u} [NormedAddCommGroup E] [NormedSpace ℂ E] (f : ℂ → E) (z w : ℂ) (s : Set ℂ)
    (hs : s.Countable) (Hc : ContinuousOn f (Set.uIcc z.re w.re ×ℂ Set.uIcc z.im w.im))
    (Hd :

     ∀ x ∈ Set.Ioo (min z.re w.re) (max z.re w.re) ×ℂ Set.Ioo (min z.im w.im) (max z.im w.im) \ s,
      DifferentiableAt ℂ f x
     )
    :
  (((∫ (x : ℝ) in z.re..w.re, f (↑x + ↑z.im * I)) - ∫ (x : ℝ) in z.re..w.re, f (↑x + ↑w.im * I)) + I •
       ∫ (y : ℝ) in z.im..w.im, f (↑w.re + ↑y * I)) - I • ∫ (y : ℝ) in z.im..w.im, f (↑z.re + ↑y * I)
       = 0
```

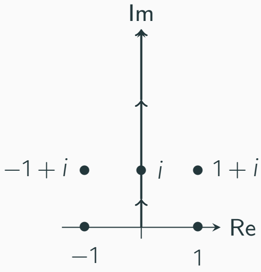
Cauchy-Goursat theorem for a rectangle: the integral of a complex differentiable function over the boundary of a rectangle equals zero. More precisely, if `f` is continuous on a closed rectangle and is complex differentiable at all but countably many points of the corresponding open rectangle, then its integral over the boundary of the rectangle equals zero.

Double Zeroes at Lattice Points

$$- 4 \sin^2\left(\frac{\pi \left\|x\right\|^2}{2}\right) \int_0^{i\infty} \phi_0\left(\frac{-1}{z}\right) z^2\, e^{\pi i \left\|x\right\|^2 z}\, dz$$

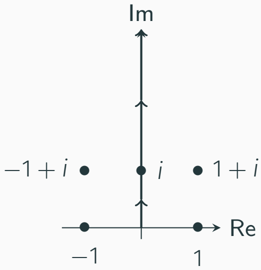
Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2\,e^{\pi i\|x\|^2 z}\,dz$$



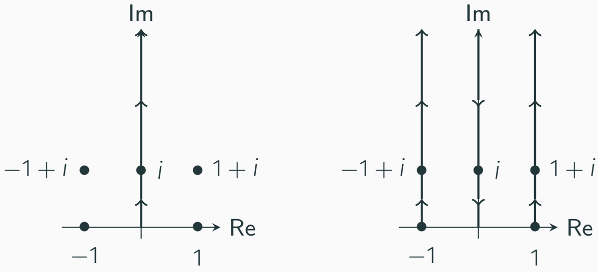
Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2\,e^{\pi i\|x\|^2 z}\,dz = \int_{-1}^{-1+i\infty} + \int_1^{1+i\infty} - 2\int_0^{i\infty}$$



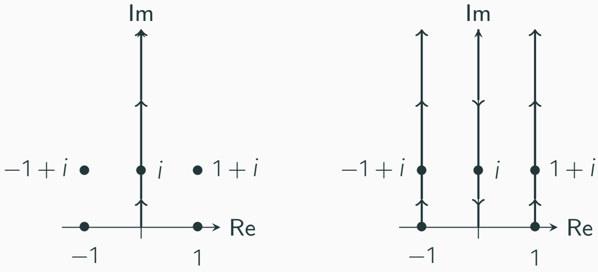
Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right) z^2\, e^{\pi i\|x\|^2 z}\,dz = \int_{-1}^{-1+i\infty} + \int_1^{1+i\infty} -2\int_0^{i\infty}$$



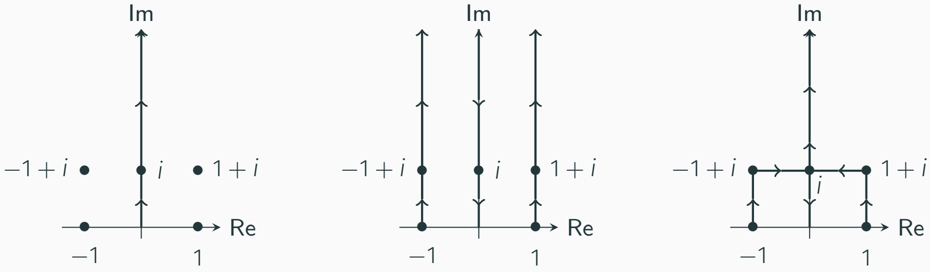
Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2\,e^{\pi i\|x\|^2 z}\,dz=\int_{-1}^{-1+i\infty}+\int_{1}^{1+i\infty}-2\int_0^{i\infty}=\int_{-1}^{i}+\int_{1}^{i}-2\int_0^{i}+2\int_i^{i\infty}$$



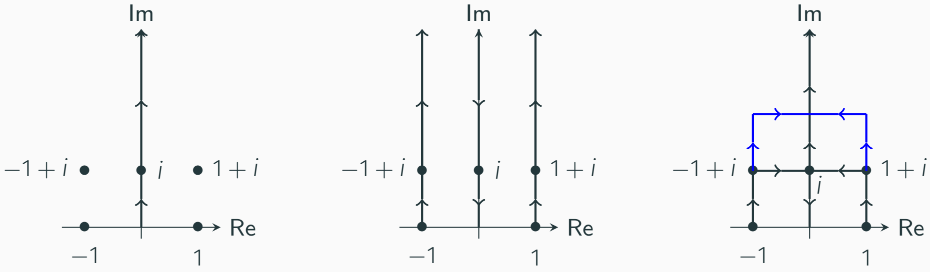
Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\left\|x\right\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2\,e^{\pi i\|x\|^2 z}\,dz=\int_{-1}^{-1+i\infty}+\int_{1}^{1+i\infty}-2\int_0^{i\infty}=\int_{-1}^{i}+\int_{1}^{i}-2\int_0^{i}+2\int_i^{i\infty}$$

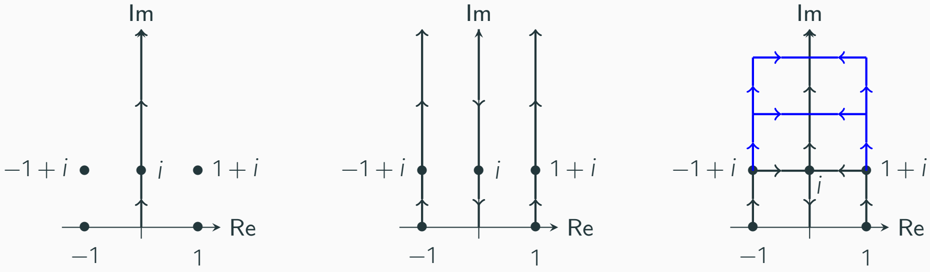


Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2e^{\pi i\|x\|^2 z}\,dz=\int_{-1}^{-1+i\infty}+\int_1^{1+i\infty}-2\int_0^{i\infty}=\int_{-1}^{i}+\int_1^{i}-2\int_0^{i}+2\int_i^{i\infty}$$

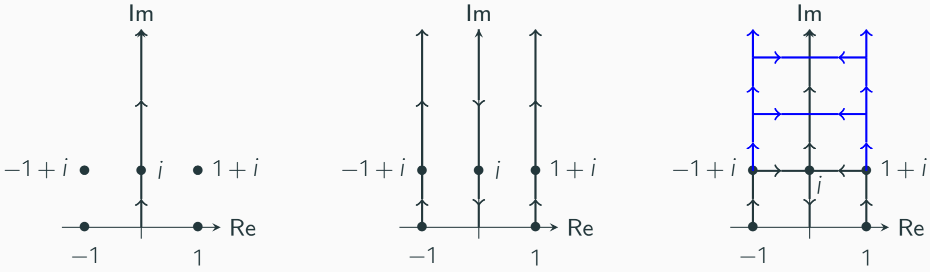


$$-4\sin^2\left(\frac{\pi\,\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2\,e^{\pi i\|x\|^2 z}\,dz=\int_{-1}^{-1+i\infty}+\int_1^{1+i\infty}-2\int_0^{i\infty}=\int_{-1}^{i}+\int_1^{i}-2\int_0^{i}+2\int_i^{i\infty}$$



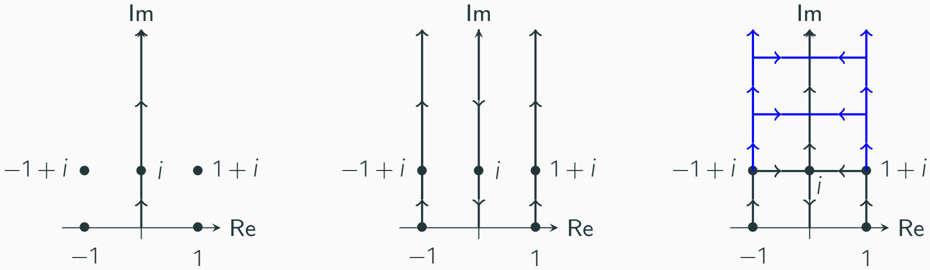
Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\,\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2\,e^{\pi i\|x\|^2 z}\,dz=\int_{-1}^{-1+i\infty}+\int_1^{1+i\infty}-2\int_0^{i\infty}=\int_{-1}^{i}+\int_1^{i}-2\int_0^{i}+2\int_i^{i\infty}$$



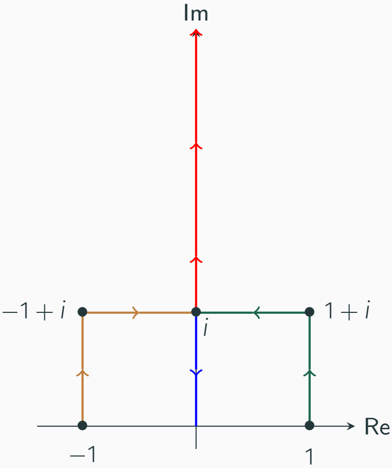
Double Zeroes at Lattice Points

$$-4\sin^2\left(\frac{\pi\,\|x\|^2}{2}\right)\int_0^{i\infty}\phi_0\left(\frac{-1}{z}\right)z^2\,e^{\pi i\|x\|^2 z}\,dz = \int_{-1}^{-1+i\infty} + \int_1^{1+i\infty} - 2\int_0^{i\infty} = \int_{-1}^{i} + \int_1^{i} - 2\int_0^{i} + 2\int_i^{i\infty} = a(x)$$

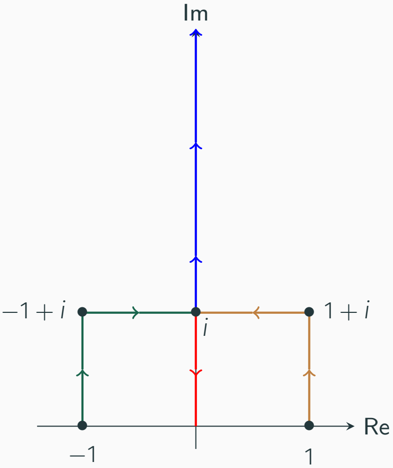


The Fourier Eigenfunction Property

Before taking the Fourier transform:

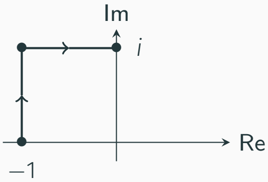


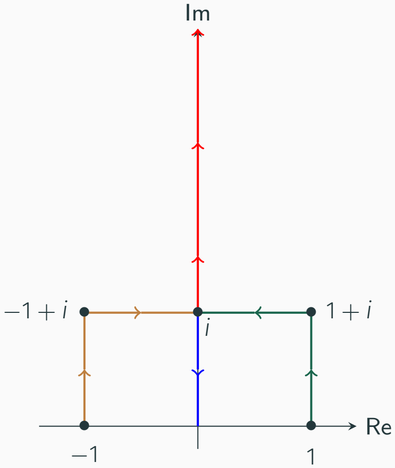
After taking the Fourier transform:



The Fourier Eigenfunction Property

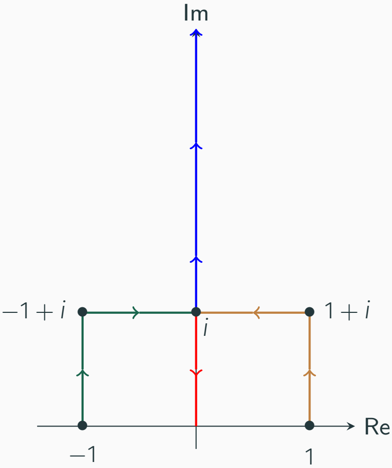
Before taking the Fourier transform:

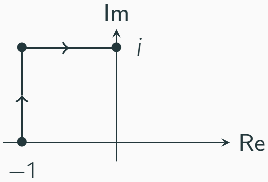


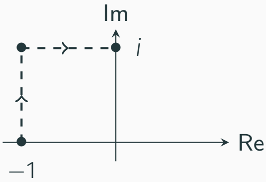


The Fourier Eigenfunction Property

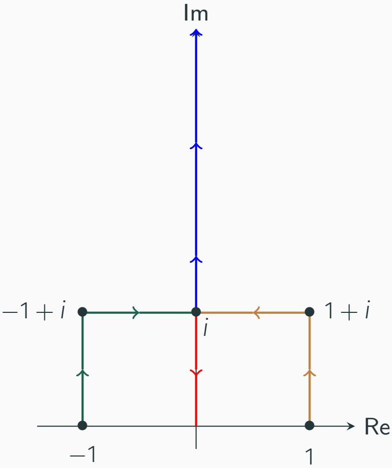
After taking the Fourier transform:

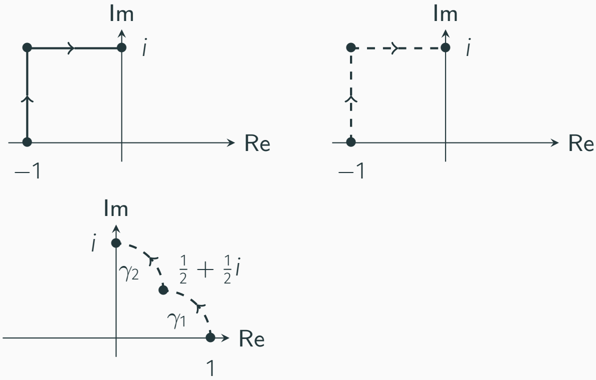






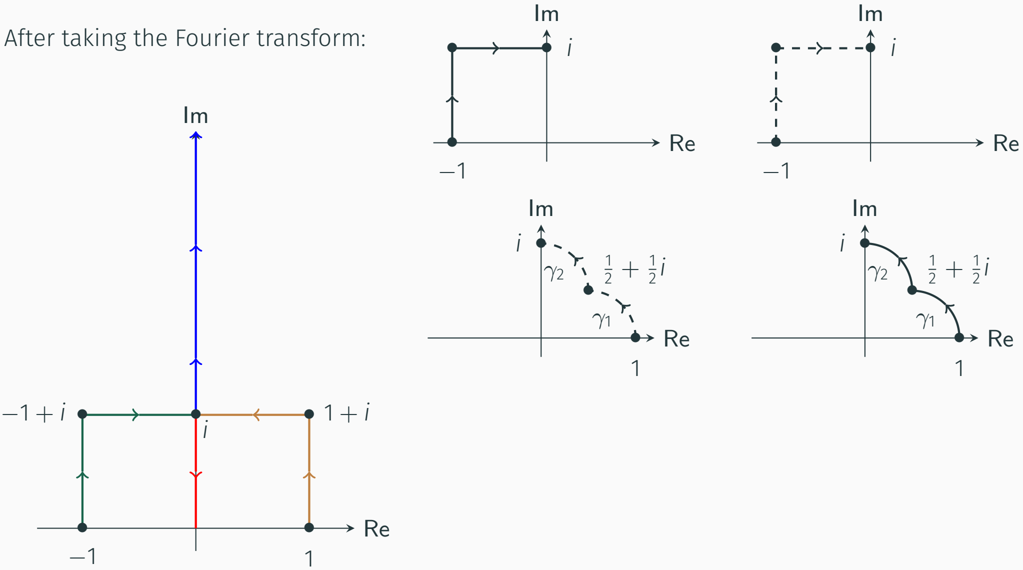
After taking the Fourier transform:





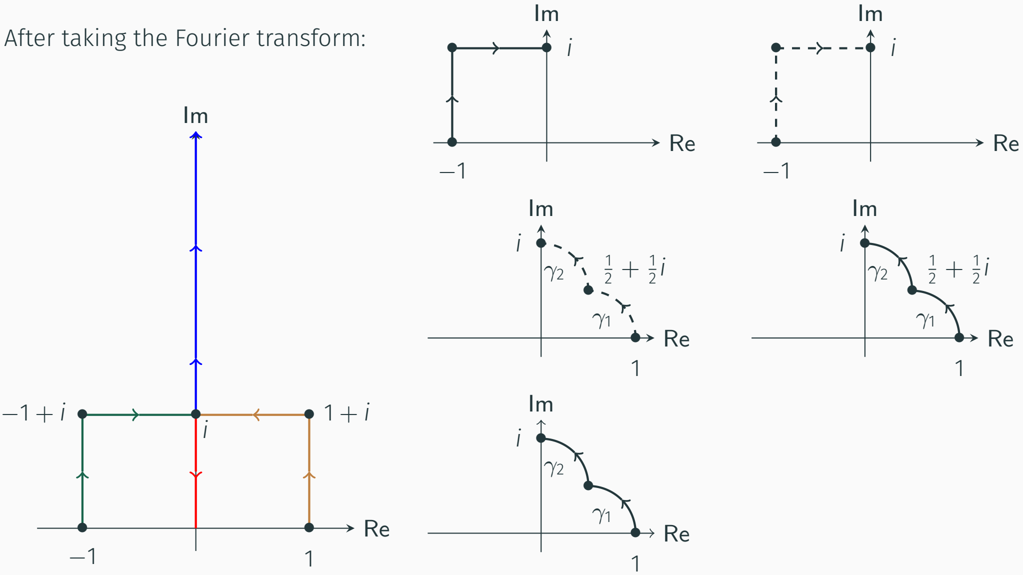
The Fourier Eigenfunction Property

After taking the Fourier transform:



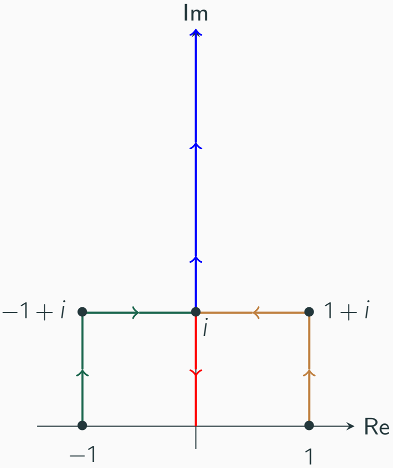
The Fourier Eigenfunction Property

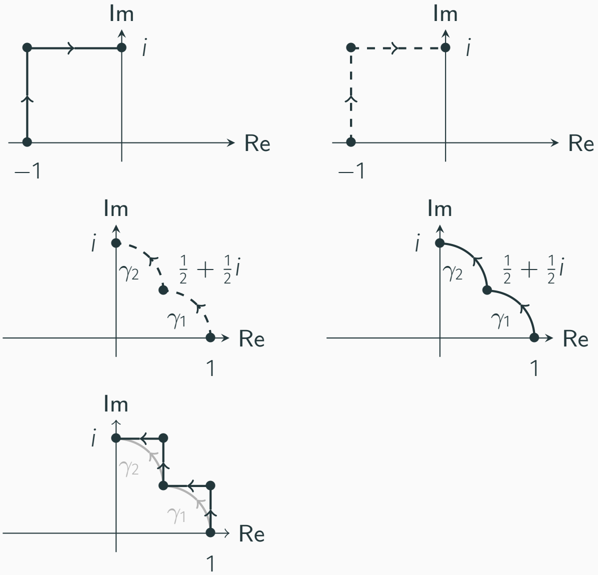
After taking the Fourier transform:



The Fourier Eigenfunction Property

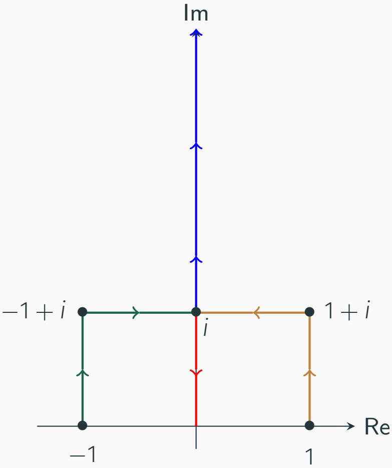
After taking the Fourier transform:

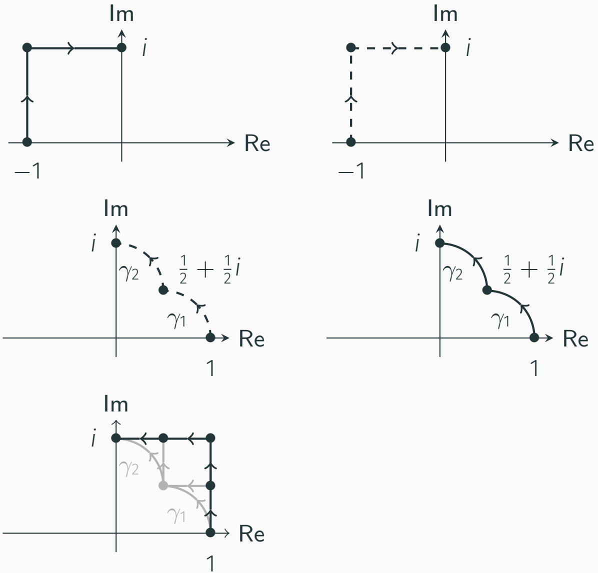




The Fourier Eigenfunction Property

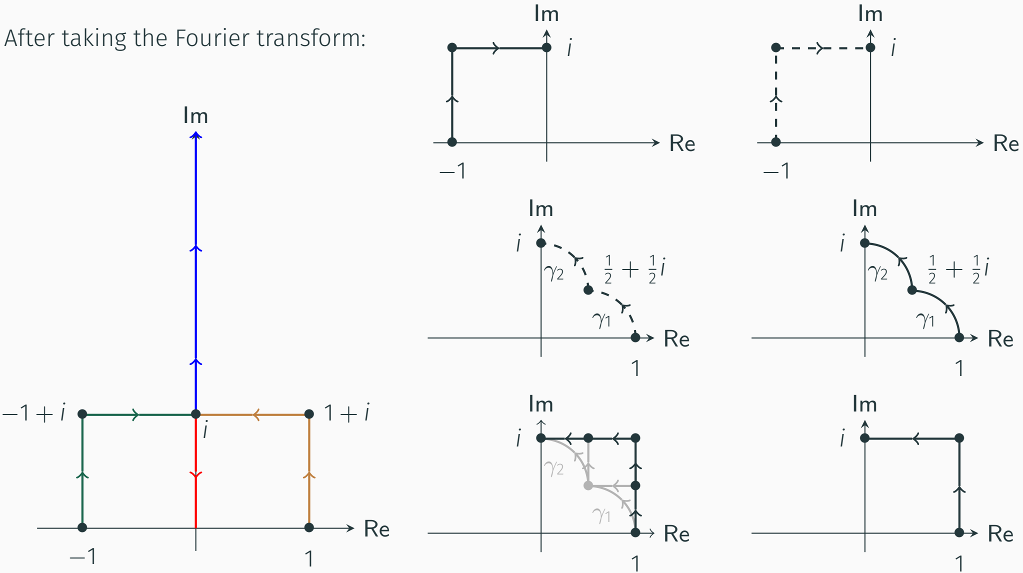
After taking the Fourier transform:





The Fourier Eigenfunction Property

After taking the Fourier transform:



Complex Computations are Complex

Metaprogramming and Complex Numbers

Computations in Lean are difficult.

Computations in Lean are difficult. We have **tactics** to help us:

Computations in Lean are difficult. We have **tactics** to help us: `simp`

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp`

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals.

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals.

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral.

Metaprogramming and Complex Numbers

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea:

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well.

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well. We can extend `norm_num` to handle such expressions.

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, $4/2$, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well. We can extend `norm_num` to handle such expressions.

For example, $\left(\overline{3+4i}\right)(3+4i) = 25 + 0i$.

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well. We can extend `norm_num` to handle such expressions.

For example, $\left(\overline{3+4i}\right)(3+4i) = 25 + 0i$. If we separate $\left(\overline{3+4i}\right)(3+4i)$ into real and imaginary parts, these can be simplified by `norm_num` to 25 and 0 respectively.

Metaprogramming and Complex Numbers

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well. We can extend `norm_num` to handle such expressions.

For example, $\left(\overline{3+4i}\right)(3+4i) = 25 + 0i$. If we separate $\left(\overline{3+4i}\right)(3+4i)$ into real and imaginary parts, these can be simplified by `norm_num` to 25 and 0 respectively.

```
example : (conj (3 + 4 * I) : ℂ) * (3 + 4 * I) = 25 := calc
  _ = (3 - 4 * I) * (3 + 4 * I) := by simp [conj_ofNat, sub_eq_add_neg]
  _ = _ := by
    ring_nf
    simp only [I_sq, neg_mul, one_mul, sub_neg_eq_add]
    norm_cast
    -- The old `norm_num` would also have done the last step but not any of the previous ones
```

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well. We can extend `norm_num` to handle such expressions.

For example, $\left(\overline{3+4i}\right)(3+4i) = 25 + 0i$. If we separate $\left(\overline{3+4i}\right)(3+4i)$ into real and imaginary parts, these can be simplified by `norm_num` to 25 and 0 respectively.

```
example : (conj (3 + 4 * I) : ℂ) * (3 + 4 * I) = 25 := by norm_num
```

Metaprogramming and Complex Numbers

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well. We can extend `norm_num` to handle such expressions.

For example, $\left(\overline{3+4i}\right)(3+4i) = 25 + 0i$. If we separate $\left(\overline{3+4i}\right)(3+4i)$ into real and imaginary parts, these can be simplified by `norm_num` to 25 and 0 respectively.

```
example : (conj (3 + 4 * I) : ℂ) * (3 + 4 * I) = 25 := by norm_num
```

This automation was jointly developed with Heather Macbeth and Edison Xie.

Metaprogramming and Complex Numbers

Computations in Lean are difficult. We have **tactics** to help us: `simp` , `field_simp` , `norm_num`.

`norm_num` simplifies numerals. 1, 4/2, 1.1 are all numerals. π is not a numeral. i is not a numeral either.

Idea: If an expression is made up of several complex numbers such that the real and imaginary part of each number is a numeral, its real and imaginary parts are numerals as well. We can extend `norm_num` to handle such expressions.

For example, $\left(\overline{3+4i}\right)(3+4i) = 25 + 0i$. If we separate $\left(\overline{3+4i}\right)(3+4i)$ into real and imaginary parts, these can be simplified by `norm_num` to 25 and 0 respectively.

```
example : (conj (3 + 4 * I) : ℂ) * (3 + 4 * I) = 25 := by norm_num
```

This automation was jointly developed with Heather Macbeth and Edison Xie. It is `mathlib` pull request #26975 on GitHub.

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise.

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements.

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability: Has finite integral and a.e. strongly measurable.

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability: Has finite integral and a.e. strongly measurable.

Not satisfying these assumptions doesn't give you an error!

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability: Has finite integral and a.e. strongly measurable.

Not satisfying these assumptions doesn't give you an error!

If the assumptions aren't satisfied, an expression using a sum/product/integral/etc. is assigned a *junk value*.

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability: Has finite integral and a.e. strongly measurable.

Not satisfying these assumptions doesn't give you an error!

If the assumptions aren't satisfied, an expression using a sum/product/integral/etc. is assigned a *junk value*. Unless one is careful, one may not immediately realise a statement is wrong!

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability: Has finite integral and a.e. strongly measurable.

Not satisfying these assumptions doesn't give you an error!

If the assumptions aren't satisfied, an expression using a sum/product/integral/etc. is assigned a *junk value*. Unless one is careful, one may not immediately realise a statement is wrong!

To avoid dealing with junk values, we can use `HasSum`.

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability: Has finite integral and a.e. strongly measurable.

Not satisfying these assumptions doesn't give you an error!

If the assumptions aren't satisfied, an expression using a sum/product/integral/etc. is assigned a *junk value*. Unless one is careful, one may not immediately realise a statement is wrong!

To avoid dealing with junk values, we can use `HasSum`. **Caveat**: every use of `HasSum` requires convergence proof.

Computations involving Functions

We have seen that metaprogramming is a powerful way of simplifying expressions involving *numbers* in $\mathbb{C}$—numerals or otherwise. Unfortunately, this doesn't overcome all the challenges of performing computations in Lean.

The biggest challenge in performing calculations involving functions is proving (and carrying around) 'compatible properties':

1. Summability: Absolute convergence of sums.
2. Multipliability: Convergence of products and all rearrangements. *Strictly weaker than absolute convergence of sum of logs: 'convergence to zero' allowed.*
3. Integrability: Has finite integral and a.e. strongly measurable.

Not satisfying these assumptions doesn't give you an error!

If the assumptions aren't satisfied, an expression using a sum/product/integral/etc. is assigned a *junk value*. Unless one is careful, one may not immediately realise a statement is wrong!

To avoid dealing with junk values, we can use `HasSum`. **Caveat**: every use of `HasSum` requires convergence proof. **Workaround**: use sum notation and only prove convergence when needed.

Questions?

[1] H. Cohn and N. Elkies.
New Upper Bounds on Sphere Packings I.
Annals of Mathematics, 157(2):689–714, 2003.

[2] T. F. Görbe.
Exceptionally Beautiful Symmetries.
`https://tamasgorbe.com/symmetry`.

[3] T. C. Hales.
Introduction to the Flyspeck Project.
In T. Coquand, H. Lombardi, and M.-F. Roy, editors, *Mathematics, Algorithms, Proofs*, volume 5021 of *Dagstuhl Seminar Proceedings (DagSemProc)*, pages 1–11, Dagstuhl, Germany, 2006. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[4] M. S. Viazovska.
The sphere packing problem in dimension 8.
Annals of Mathematics, 185(3):991–1015, 2017.