universität
innsbruck

Some new tricks for formalising advanced mathematics

Manuel Eberl

Outline

1. A quick look at Isabelle/HOL

Outline

1. A quick look at Isabelle/HOL
2. Some recent work in formalising graduate-level maths

Outline

1. A quick look at Isabelle/HOL
2. Some recent work in formalising graduate-level maths
3. Some tools and tricks we had to develop along the way

What is Isabelle/HOL?

What is Isabelle/HOL

- Isabelle: An interactive theorem prover / proof assistant (like Coq, Lean, Mizar)

What is Isabelle/HOL

- Isabelle: An interactive theorem prover / proof assistant (like Coq, Lean, Mizar)
- Can do multiple logics in principle; mostly Higher-Order Logic (HOL)

What is Isabelle/HOL

- Isabelle: An interactive theorem prover / proof assistant (like Coq, Lean, Mizar)
- Can do multiple logics in principle; mostly Higher-Order Logic (HOL)
- Focus on good automation and high performance

What is Isabelle/HOL

- Isabelle: An interactive theorem prover / proof assistant (like Coq, Lean, Mizar)
- Can do multiple logics in principle; mostly Higher-Order Logic (HOL)
- Focus on good automation and high performance
- Unlike e.g. Coq: no dependent types, no attempt to avoid classical logic

What is Isabelle/HOL

- Isabelle: An interactive theorem prover / proof assistant (like Coq, Lean, Mizar)
- Can do multiple logics in principle; mostly Higher-Order Logic (HOL)
- Focus on good automation and high performance
- Unlike e.g. Coq: no dependent types, no attempt to avoid classical logic
- Large library of mathematics (especially analysis)

What is Isabelle/HOL

- Isabelle: An interactive theorem prover / proof assistant (like Coq, Lean, Mizar)
- Can do multiple logics in principle; mostly Higher-Order Logic (HOL)
- Focus on good automation and high performance
- Unlike e.g. Coq: no dependent types, no attempt to avoid classical logic
- Large library of mathematics (especially analysis)
- Archive of Formal Proofs: Growing collection of entries on Mathematics, Computer Science, Logic; 748 articles, 457 authors

Some Examples

A typical structure Isabelle proof: If one of a and b is even, then $a \cdot b$ is even.

```
lemma
  fixes a b :: int
  assumes "even a ∨ even b"
  shows   "even (a * b)"
proof
  from assms show "even (a * b)"
  proof
    assume "even a"
    then obtain a' where a': "a = 2 * a'"

    have "even (2 * (a' * b))"

    also have "2 * (a' * b) = a * b"
      using a'
    finally show "even (a * b)"
  next
    assume "even b"
    (* ... *)
```

Some Examples

A typical structure Isabelle proof: If one of a and b is even, then $a \cdot b$ is even.

```
lemma
  fixes a b :: int
  assumes "even a ∨ even b"
  shows   "even (a * b)"
proof -
  from assms show "even (a * b)"
  proof
    assume "even a"
    then obtain a' where a': "a = 2 * a'"
      by (elim evenE)
    have "even (2 * (a' * b))"
      by simp
    also have "2 * (a' * b) = a * b"
      using a' by simp
    finally show "even (a * b)" .
  next
    assume "even b"
    (* ... *)
```

Some Examples

Dirichlet's Theorem: Given coprime integers *h* and *n*, there are infinitely many primes congruent *h* modulo *n*:

```
theorem Dirichlet:
  assumes "coprime h n"
  shows   "infinite {p. prime p ∧ [p = h] (mod n)}"
```

Some Examples

A famous equality involving the Riemann ζ function:

$$\zeta(2n) = \frac{(-1)^{n+1} B_{2n} (2\pi)^{2n}}{2(2n)!}$$

Some Examples

A famous equality involving the Riemann ζ function:

$$\zeta(2n) = \frac{(-1)^{n+1} B_{2n} (2\pi)^{2n}}{2(2n)!}$$

```
theorem zeta_even_nat:
  "zeta (2 * of_nat n) =
    of_real ((-1)^(n+1) * bernoulli (2*n) * (2*pi)^(2*n) / (2 * fact (2*n)))"
```

Some Examples

Stirling's formula: $n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

Some Examples

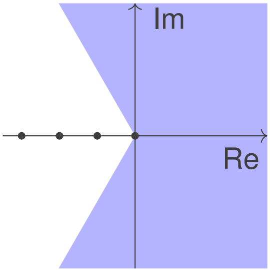
Stirling's formula: $n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

```
theorem fact_asymp_equiv:
  "fact ∼[at_top] (λn. sqrt (2*pi*n) * (n / exp 1) ^ n :: real)"
```

Some Examples

Stirling's formula for the complex Γ function:

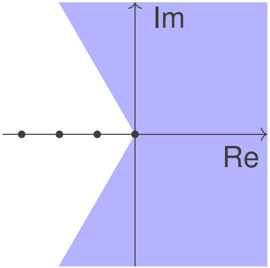
$$\Gamma(s) \sim \sqrt{\frac{2\pi}{s}}\left(\frac{s}{e}\right)^s \text{ uniformly for } |s| \to \infty \text{ with } |\mathrm{Arg}\, s| \leq \alpha < \pi$$



Some Examples

Stirling's formula for the complex Γ function:

$$\Gamma(s) \sim \sqrt{\frac{2\pi}{s}}\left(\frac{s}{e}\right)^s \text{ uniformly for } |s| \to \infty \text{ with } |\mathrm{Arg}\ s| \leq \alpha < \pi$$




```
lemma Gamma_complex_asymp_equiv:
  fixes F and α
  assumes "α ∈ {0..<pi}"
  defines "F ≡ at_infinity ⊓ principal {z. ¦Arg z¦ ≤ α}"
  shows   "Gamma ∼[F] (λs. sqrt (2 * pi / s) powr (1 / 2) * (s / exp 1) powr s)"
```

Some Examples

The “Sophomore’s Dream”: $\displaystyle\int_0^1 x^{-x}\,\mathrm{d}x = \sum_{k=1}^{\infty} k^{-k}$

Some Examples

The "Sophomore's Dream": $\int_0^1 x^{-x}\,dx = \sum_{k=1}^{\infty} k^{-k}$

```
theorem sophomores_dream1:
  "integral {0..1} (λx. x powr (-x)) = (∑∞ k∈{(1::nat)..}. k powi (-k))"
```

A look back

Formalising mathematics tends to be a very tedious and exhausting process, so I think it is worth stressing how far we have come:

A look back

Formalising mathematics tends to be a very tedious and exhausting process, so I think it is worth stressing how far we have come:

- 40 years ago, formalising the analytic proof of the Prime Number Theorem was a distant dream.
  Larry Paulson: “I don’t know that I expected to see it in my lifetime.”

A look back

Formalising mathematics tends to be a very tedious and exhausting process, so I think it is worth stressing how far we have come:

- 40 years ago, formalising the analytic proof of the Prime Number Theorem was a distant dream.
  Larry Paulson: “I don’t know that I expected to see it in my lifetime.”
- 20 years ago it still seemed like an extremely daunting task:
  Robert Solovay: “It will take decades until ITPs are up to the task.”

A look back

Formalising mathematics tends to be a very tedious and exhausting process, so I think it is worth stressing how far we have come:

- 40 years ago, formalising the analytic proof of the Prime Number Theorem was a distant dream.
  Larry Paulson: “I don’t know that I expected to see it in my lifetime.”
- 20 years ago it still seemed like an extremely daunting task:
  Robert Solovay: “It will take decades until ITPs are up to the task.”
- 15 years ago, John Harrison did it in HOL Light (for fun).

A look back

Formalising mathematics tends to be a very tedious and exhausting process, so I think it is worth stressing how far we have come:

- 40 years ago, formalising the analytic proof of the Prime Number Theorem was a distant dream.
  Larry Paulson: "I don't know that I expected to see it in my lifetime."
- 20 years ago it still seemed like an extremely daunting task:
  Robert Solovay: "It will take decades until ITPs are up to the task."
- 15 years ago, John Harrison did it in HOL Light (for fun).
- 5 years ago, I formalised an entire textbook (for fun), with PNT as a part of it.

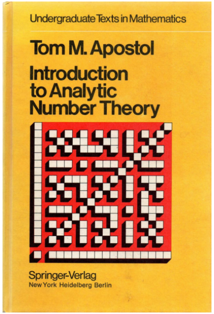
A look back

Formalising mathematics tends to be a very tedious and exhausting process, so I think it is worth stressing how far we have come:

- 40 years ago, formalising the analytic proof of the Prime Number Theorem was a distant dream.
  Larry Paulson: “I don’t know that I expected to see it in my lifetime.”
- 20 years ago it still seemed like an extremely daunting task:
  Robert Solovay: “It will take decades until ITPs are up to the task.”
- 15 years ago, John Harrison did it in HOL Light (for fun).
- 5 years ago, I formalised an entire textbook (for fun), with PNT as a part of it.
- Now we are tackling graduate-level maths (elliptic functions, modular forms).

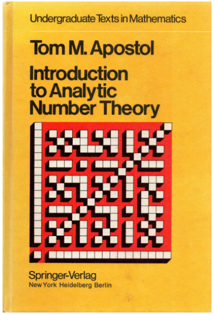
A look back

(Almost) an entire undergraduate maths textbook formalised



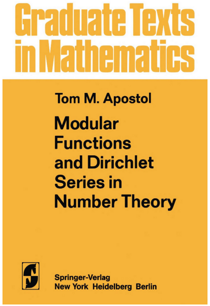
A look back

(Almost) an entire undergraduate maths textbook formalised

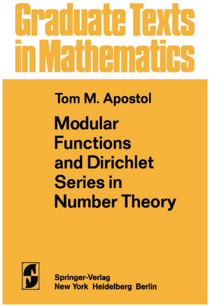


Contents: Dirichlet series, characters, Gauss sums, Riemann ζ function, L functions, lots of facts about prime numbers

Current work

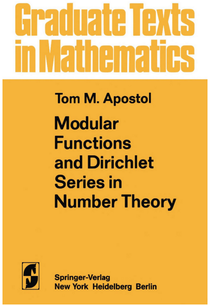


Current work



About 50 % formalised.

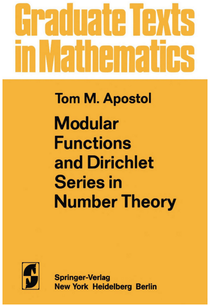
Current work



About 50 % formalised.

Already done: Elliptic functions, Klein's j function, basic properties of modular forms, Dedekind's η function

Current work



About 50 % formalised.

Already done: Elliptic functions, Klein's j function, basic properties of modular forms, Dedekind's η function

Up next: Rademacher's formula for the partition function, advanced results about modular forms

Current work

Formalising graduate-level mathematics like this is challenging:

- builds on a great body of underlying mathematics (e.g. complex analysis)
  If your library does not have all of it, you're out of luck

Current work

Formalising graduate-level mathematics like this is challenging:

- builds on a great body of underlying mathematics (e.g. complex analysis)
  If your library does not have all of it, you're out of luck
- non-trivial results from other sources are only referenced instead of proven

Current work

Formalising graduate-level mathematics like this is challenging:

- builds on a great body of underlying mathematics (e.g. complex analysis)
  If your library does not have all of it, you're out of luck
- non-trivial results from other sources are only referenced instead of proven
- great liberties with notation are taken, proofs left very vague, cases and side conditions omitted

Current work

Formalising graduate-level mathematics like this is challenging:

- builds on a great body of underlying mathematics (e.g. complex analysis)
  If your library does not have all of it, you're out of luck
- non-trivial results from other sources are only referenced instead of proven
- great liberties with notation are taken, proofs left very vague, cases and side conditions omitted
- subtle issues such as cancellation of poles and zeros are ignored

Current work

Formalising graduate-level mathematics like this is challenging:

- builds on a great body of underlying mathematics (e.g. complex analysis)
  If your library does not have all of it, you're out of luck
- non-trivial results from other sources are only referenced instead of proven
- great liberties with notation are taken, proofs left very vague, cases and side conditions omitted
- subtle issues such as cancellation of poles and zeros are ignored

Sometimes page after page can be formalised in a straightfoward way.

Current work

Formalising graduate-level mathematics like this is challenging:

- builds on a great body of underlying mathematics (e.g. complex analysis)
  If your library does not have all of it, you're out of luck
- non-trivial results from other sources are only referenced instead of proven
- great liberties with notation are taken, proofs left very vague, cases and side conditions omitted
- subtle issues such as cancellation of poles and zeros are ignored

Sometimes page after page can be formalised in a straightfoward way.
But sometimes a single line causes difficulties for days.

Current work

Formalising graduate-level mathematics like this is challenging:

- builds on a great body of underlying mathematics (e.g. complex analysis)
  If your library does not have all of it, you're out of luck
- non-trivial results from other sources are only referenced instead of proven
- great liberties with notation are taken, proofs left very vague, cases and side conditions omitted
- subtle issues such as cancellation of poles and zeros are ignored

Sometimes page after page can be formalised in a straightfoward way.
But sometimes a single line causes difficulties for days.

But my impression is that things are getting better.

Current work

The analytic proof of the PNT seemed impossible 30 years ago, very daunting (but doable) 15 years ago. 5 years ago it was routine and today it is almost trivial!

Current work

The analytic proof of the PNT seemed impossible 30 years ago, very daunting (but doable) 15 years ago. 5 years ago it was routine and today it is almost trivial!

Why did things get so much easier? Good libraries and good tool support

Current work

The analytic proof of the PNT seemed impossible 30 years ago, very daunting (but doable) 15 years ago. 5 years ago it was routine and today it is almost trivial!

Why did things get so much easier? Good libraries and good tool support

I will mention four such tools in particular:

Current work

The analytic proof of the PNT seemed impossible 30 years ago, very daunting (but doable) 15 years ago. 5 years ago it was routine and today it is almost trivial!

Why did things get so much easier? Good libraries and good tool support

I will mention four such tools in particular:

- Multiseries expansion

Current work

The analytic proof of the PNT seemed impossible 30 years ago, very daunting (but doable) 15 years ago. 5 years ago it was routine and today it is almost trivial!

Why did things get so much easier? Good libraries and good tool support

I will mention four such tools in particular:

- Multiseries expansion
- Laurent series expansions

Current work

The analytic proof of the PNT seemed impossible 30 years ago, very daunting (but doable) 15 years ago. 5 years ago it was routine and today it is almost trivial!

Why did things get so much easier? Good libraries and good tool support

I will mention four such tools in particular:

- Multiseries expansion
- Laurent series expansions
- Evaluation of winding numbers

Current work

The analytic proof of the PNT seemed impossible 30 years ago, very daunting (but doable) 15 years ago. 5 years ago it was routine and today it is almost trivial!

Why did things get so much easier? Good libraries and good tool support

I will mention four such tools in particular:

- Multiseries expansion
- Laurent series expansions
- Evaluation of winding numbers
- Deformation of integration contours

Multiseries expansions

“Obvious” Limits

“Routine” limit problems are ubiquitous and mathematicians often do not even bother giving proofs for them:

“Obvious” Limits

“Routine” limit problems are ubiquitous and mathematicians often do not even bother giving proofs for them:

- For any $y > 0$, the following function is bounded for all $x \in \mathbb{R}$:

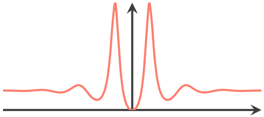
$$\frac{\sin^2(\frac{1}{2}x) + \sinh^2(\frac{1}{2}xy)}{\cosh(xy) + \cos(x)}$$

"Obvious" Limits

"Routine" limit problems are ubiquitous and mathematicians often do not even bother giving proofs for them:

- For any $y > 0$, the following function is bounded for all $x \in \mathbb{R}$:

$$\frac{\sin^2(\frac{1}{2}x) + \sinh^2(\frac{1}{2}xy)}{\cosh(xy) + \cos(x)}$$

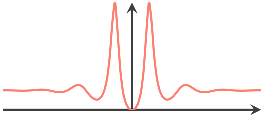


"Obvious" Limits

"Routine" limit problems are ubiquitous and mathematicians often do not even bother giving proofs for them:

- For any $y > 0$, the following function is bounded for all $x \in \mathbb{R}$:

$$\frac{\sin^2(\frac{1}{2}x) + \sinh^2(\frac{1}{2}xy)}{\cosh(xy) + \cos(x)}$$



- For any $n \in \mathbb{N}$, the series

$$\sum_{k=1}^{\infty} \left(\frac{\log(k)^n}{k} - \frac{\log(k+1)^{n+1} - \log(k)^{n+1}}{n+1} \right)$$

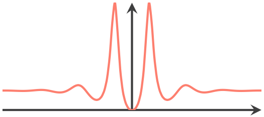
converges

"Obvious" Limits

"Routine" limit problems are ubiquitous and mathematicians often do not even bother giving proofs for them:

- For any $y > 0$, the following function is bounded for all $x \in \mathbb{R}$:

$$\frac{\sin^2(\frac{1}{2}x) + \sinh^2(\frac{1}{2}xy)}{\cosh(xy) + \cos(x)}$$



- For any $n \in \mathbb{N}$, the series

$$\sum_{k=1}^{\infty} \left(\frac{\log(k)^n}{k} - \frac{\log(k+1)^{n+1} - \log(k)^{n+1}}{n+1} \right)$$

converges due to the comparison test, since the summand is $\sim k^{-2} \log(k)^{n-1}$.

“Obvious” Limits

Proving even much simpler asymptotic estimates in a theorem prover is a mind-numbingly tedious exercise.

“Obvious” Limits

Proving even much simpler asymptotic estimates in a theorem prover is a mind-numbingly tedious exercise. (trust me on this)

“Obvious” Limits

Proving even much simpler asymptotic estimates in a theorem prover is a mind-numbingly tedious exercise. (trust me on this)

But if you type them into Wolfram Alpha, you often get an answer (usually even the right one).

“Obvious” Limits

Proving even much simpler asymptotic estimates in a theorem prover is a mind-numbingly tedious exercise. (trust me on this)

But if you type them into Wolfram Alpha, you often get an answer (usually even the right one).

How do computer algebra systems achieve this?

“Obvious” Limits

Proving even much simpler asymptotic estimates in a theorem prover is a mind-numbingly tedious exercise. (trust me on this)

But if you type them into Wolfram Alpha, you often get an answer (usually even the right one).

How do computer algebra systems achieve this?

By computing Multiseries expansions (or something very similar)!

The `real_asymp` Method

I looked through the papers from the 90s that explained how these algorithms work and implemented them in Isabelle.

The `real_asymp` Method

I looked through the papers from the 90s that explained how these algorithms work and implemented them in Isabelle.

This is code is proof-producing: You do not need to trust it. To create theorems, it has to go through the Isabelle kernel.

The `real_asymp` Method

I looked through the papers from the 90s that explained how these algorithms work and implemented them in Isabelle.

This is code is proof-producing: You do not need to trust it. To create theorems, it has to go through the Isabelle kernel.

Applicable to a big class of real-valued functions consisting of arithmetic, $\log$, $\exp$, trigonometric functions, absolute value, . . .

The `real_asymp` Method

I looked through the papers from the 90s that explained how these algorithms work and implemented them in Isabelle.

This is code is proof-producing: You do not need to trust it. To create theorems, it has to go through the Isabelle kernel.

Applicable to a big class of real-valued functions consisting of arithmetic, $\log$, $\exp$, trigonometric functions, absolute value, ...

Not competitive with Mathematica/Maple in scope or performance.

The `real_asymp` Method

I looked through the papers from the 90s that explained how these algorithms work and implemented them in Isabelle.

This is code is proof-producing: You do not need to trust it. To create theorems, it has to go through the Isabelle kernel.

Applicable to a big class of real-valued functions consisting of arithmetic, $\log$, $\exp$, trigonometric functions, absolute value, ...

Not competitive with Mathematica/Maple in scope or performance.

But good enough to help with all asymptotic problems I encountered in "real-world" formalisation.

Example

```
lemma
  fixes f :: "real ⇒ real" and y :: real
  assumes "y > 0"
  defines "f ≡ (λx. (cos (x/2) ^ 2 + sinh (x*y/2) ^ 2) / (cosh (x*y) - cos x))"
  shows   "f ∈ O[at_bot](λ_. 1)" and "f ∈ O[at_top](λ_. 1)"
```

Example

```
lemma
  fixes f :: "real ⇒ real" and y :: real
  assumes "y > 0"
  defines "f ≡ (λx. (cos (x/2) ^ 2 + sinh (x*y/2) ^ 2) / (cosh (x*y) - cos x))"
  shows   "f ∈ O[at_bot](λ_. 1)" and "f ∈ O[at_top](λ_. 1)"
  using ‹y > 0› unfolding f_def by real_asymp+
```

Work in Progress: Laurent series expansions

Complex limits are (as far as I know) harder: many possible directions of approach, branch cuts, . . .

Work in Progress: Laurent series expansions

Complex limits are (as far as I know) harder: many possible directions of approach, branch cuts, . . .

But one important class of complex functions are actually easy: meromorphic functions

Work in Progress: Laurent series expansions

Complex limits are (as far as I know) harder: many possible directions of approach, branch cuts, ...

But one important class of complex functions are actually easy: meromorphic functions

- meromorphic ↔ differentiable everywhere except for a discrete set of poles.

Work in Progress: Laurent series expansions

Complex limits are (as far as I know) harder: many possible directions of approach, branch cuts, . . .

But one important class of complex functions are actually easy: meromorphic functions

- meromorphic $\leftrightarrow$ differentiable everywhere except for a discrete set of poles.
- A meromorphic function $f : \mathbb{C} \to \mathbb{C}$ can locally be expanded as $f(z) = \sum_{n \geq n_0} a_n (z - z_0)^n$ for some $n_0 \in \mathbb{Z}$

Work in Progress: Laurent series expansions

Complex limits are (as far as I know) harder: many possible directions of approach, branch cuts, ...

But one important class of complex functions are actually easy:
meromorphic functions

- meromorphic ↔ differentiable everywhere except for a discrete set of poles.
- A meromorphic function $f : \mathbb{C} \to \mathbb{C}$ can locally be expanded as $f(z) = \sum_{n \geq n_0} a_n (z - z_0)^n$ for some $n_0 \in \mathbb{Z}$
- The exact same techniques that we used for `real_asymp` still work, but many things are actually easier.

Work in Progress: Laurent series expansions

Complex limits are (as far as I know) harder: many possible directions of approach, branch cuts, . . .

But one important class of complex functions are actually easy:
meromorphic functions

- meromorphic ↔ differentiable everywhere except for a discrete set of poles.
- A meromorphic function $f : \mathbb{C} \to \mathbb{C}$ can locally be expanded as $f(z) = \sum_{n \geq n_0} a_n (z - z_0)^n$ for some $n_0 \in \mathbb{Z}$
- The exact same techniques that we used for `real_asymp` still work, but many things are actually easier.

The theory is all there, but a lot of code has to be written to make it nicely usable.

Application: Elliptic Functions

Definition

Let ω_1, ω_2 be two complex numbers. These span a lattice $\Lambda = \{a\omega_1 + b\omega_2 \mid a, b \in \mathbb{Z}\}$.

Application: Elliptic Functions

Definition

Let ω_1, ω_2 be two complex numbers. These span a lattice $\Lambda = \{a\omega_1 + b\omega_2 \mid a, b \in \mathbb{Z}\}$.

- A meromorphic function f is called *elliptic* if $f(z+\omega) = f(z)$ for any $\omega \in \Lambda$.

Application: Elliptic Functions

Definition

Let ω_1, ω_2 be two complex numbers. These span a lattice $\Lambda = \{a\omega_1 + b\omega_2 \mid a, b \in \mathbb{Z}\}$.

- A meromorphic function f is called *elliptic* if $f(z+\omega) = f(z)$ for any $\omega \in \Lambda$.
- $\wp(z) = \frac{1}{z^2} + \sum_{\omega \in \Lambda \setminus \{0\}} \left(\frac{1}{(z-\omega)^2} - \frac{1}{\omega^2} \right)$ is the “simplest” non-constant elliptic function.

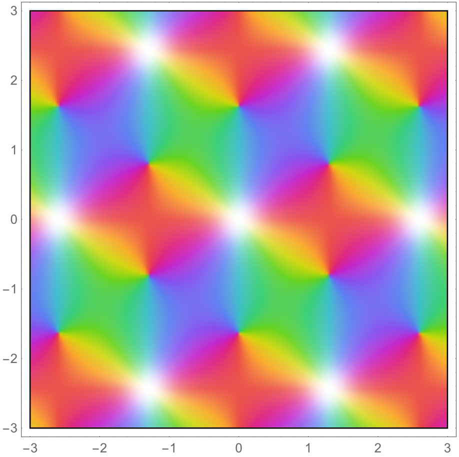
Application: Elliptic Functions

Definition

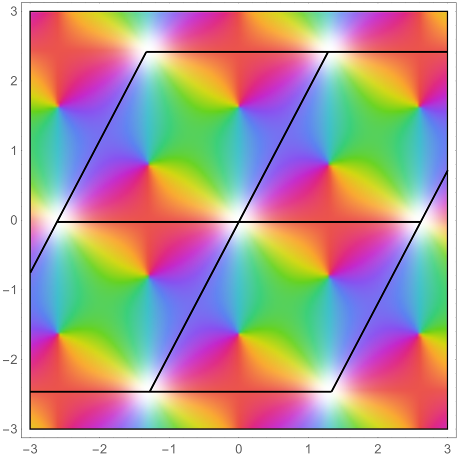
Let ω_1, ω_2 be two complex numbers. These span a lattice $\Lambda = \{a\omega_1 + b\omega_2 \mid a, b \in \mathbb{Z}\}$.

- A meromorphic function f is called *elliptic* if $f(z+\omega) = f(z)$ for any $\omega \in \Lambda$.
- $\wp(z) = \frac{1}{z^2} + \sum_{\omega \in \Lambda \setminus \{0\}} \left(\frac{1}{(z-\omega)^2} - \frac{1}{\omega^2} \right)$ is the "simplest" non-constant elliptic function.
- It has the Laurent series expansion $\wp(z) = z^{-2} + \sum_{n=2}^{\infty} (n+1) G_{n+2} z^n$ at $z = 0$.

Application: Elliptic Functions



Application: Elliptic Functions



Application: Elliptic Functions

Theorem

$$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$$

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$. Compute the Laurent expansion of g and find that all the poles cancel.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$.
Compute the Laurent expansion of g and find that all the poles cancel.
Because g is elliptic, this means that the image of g is bounded.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$.
Compute the Laurent expansion of g and find that all the poles cancel.
Because g is elliptic, this means that the image of g is bounded.
By Liouville's Theorem, g is then constant.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$.
Compute the Laurent expansion of g and find that all the poles cancel.
Because g is elliptic, this means that the image of g is bounded.
By Liouville's Theorem, g is then constant.
That constant $-140G_6$ can be read off from the Laurent series expansion.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$.
Compute the Laurent expansion of g and find that all the poles cancel.
Because g is elliptic, this means that the image of g is bounded.
By Liouville's Theorem, g is then constant.
That constant $-140G_6$ can be read off from the Laurent series expansion. □

The proof is trivial – we only have to compute a Laurent series expansion.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$.
Compute the Laurent expansion of g and find that all the poles cancel.
Because g is elliptic, this means that the image of g is bounded.
By Liouville's Theorem, g is then constant.
That constant $-140G_6$ can be read off from the Laurent series expansion. □

The proof is trivial – we only have to compute a Laurent series expansion.

But doing this in Isabelle requires a good library for reasoning about poles, Laurent series, meromorphicity, etc.

Application: Elliptic Functions

Theorem

$\wp'(z)^2 = 4\wp(z)^3 - 60G_4\wp(z) - 140G_6$

Proof.

Define $g(z) = \wp'(z)^2 - 4\wp(z)^3 + 60G_4\wp(z)$. Our goal is to show that $g(z) = -140G_6$.
Compute the Laurent expansion of g and find that all the poles cancel.
Because g is elliptic, this means that the image of g is bounded.
By Liouville's Theorem, g is then constant.
That constant $-140G_6$ can be read off from the Laurent series expansion. □

The proof is trivial – we only have to compute a Laurent series expansion.

But doing this in Isabelle requires a good library for reasoning about poles, Laurent series, meromorphicity, etc. Which we now have!

Evaluating Winding Numbers

Winding Numbers

In complex analysis one integrates along contours or “paths”, i.e. smooth functions $[0,1] \to \mathbb{C}$.

Winding Numbers

In complex analysis one integrates along contours or “paths”, i.e. smooth functions $[0,1] \to \mathbb{C}$.

Many results require us to

- focus on a certain set of “points of interest”, e.g. the singularities or zeros of some function $f : \mathbb{C} \to \mathbb{C}$

Winding Numbers

In complex analysis one integrates along contours or “paths”, i.e. smooth functions $[0,1] \to \mathbb{C}$.

Many results require us to

- focus on a certain set of “points of interest”, e.g. the singularities or zeros of some function $f : \mathbb{C} \to \mathbb{C}$
- determine the winding number of some contour γ around each such point z

Winding Numbers

In complex analysis one integrates along contours or “paths”, i.e. smooth functions $[0,1] \to \mathbb{C}$.

Many results require us to

- focus on a certain set of “points of interest”, e.g. the singularities or zeros of some function $f : \mathbb{C} \to \mathbb{C}$
- determine the winding number of some contour γ around each such point z i.e. how often does γ wind around z and in what direction

Winding Numbers

In complex analysis one integrates along contours or “paths”, i.e. smooth functions $[0,1] \to \mathbb{C}$.

Many results require us to

- focus on a certain set of “points of interest”, e.g. the singularities or zeros of some function $f : \mathbb{C} \to \mathbb{C}$
- determine the winding number of some contour γ around each such point z i.e. how often does γ wind around z and in what direction

We are mainly interested in simple closed counter-clock-wise curves.
i.e. all points have winding number 1 or 0

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Even for relatively simple contours, proving winding numbers is very annoying.

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Even for relatively simple contours, proving winding numbers is very annoying. Luckily, Wenda Li developed a tool based on Cauchy Indices that

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Even for relatively simple contours, proving winding numbers is very annoying. Luckily, Wenda Li developed a tool based on Cauchy Indices that

- works on a proof obligation of the form `winding_number` $\gamma\ z = c$

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Even for relatively simple contours, proving winding numbers is very annoying. Luckily, Wenda Li developed a tool based on Cauchy Indices that

- works on a proof obligation of the form $\texttt{winding\_number}\ \gamma\ z = c$
- where γ is a concatenation of straight lines and circular arcs

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Even for relatively simple contours, proving winding numbers is very annoying. Luckily, Wenda Li developed a tool based on Cauchy Indices that

- works on a proof obligation of the form `winding_number` $\gamma\ z = c$
- where γ is a concatenation of straight lines and circular arcs
- and reduces the proof obligation to (mostly) algebraic statements

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Even for relatively simple contours, proving winding numbers is very annoying. Luckily, Wenda Li developed a tool based on Cauchy Indices that

- works on a proof obligation of the form `winding_number` $\gamma\ z = c$
- where γ is a concatenation of straight lines and circular arcs
- and reduces the proof obligation to (mostly) algebraic statements
- which can often be solved using Isabelle's general purpose automation

Winding Numbers

Determining winding numbers is intuitively obvious: you draw the curve on paper and you can *see* what the winding number is.

Unfortunately, theorem provers do not have intuition.

Even for relatively simple contours, proving winding numbers is very annoying. Luckily, Wenda Li developed a tool based on Cauchy Indices that

- works on a proof obligation of the form $\texttt{winding\_number}\ \gamma\ z = c$
- where γ is a concatenation of straight lines and circular arcs
- and reduces the proof obligation to (mostly) algebraic statements
- which can often be solved using Isabelle's general purpose automation

Definitely *not* a fully automatic tool, but it does help immensely.

Deforming Integration Contours

Deforming Integration Contours

Lastly, I will talk in some more detail about the latest addition to our bag of tricks.

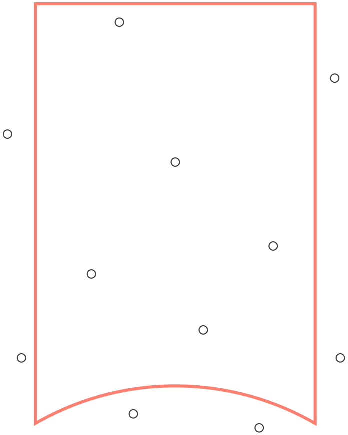
Deforming Integration Contours

Lastly, I will talk in some more detail about the latest addition to our bag of tricks.

This concerns another common problem encountered when formalising a complex-analytic argument:

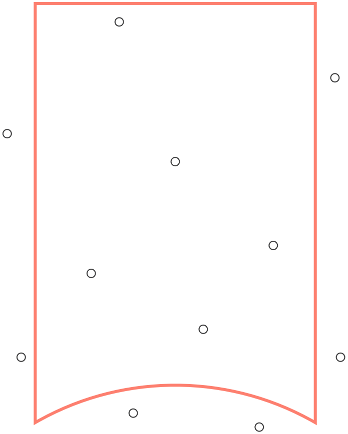
Deforming an integration contour in order to avoid “bad” points.

Counting Points



The following is a problem that arises in proving the valence formula for modular functions.

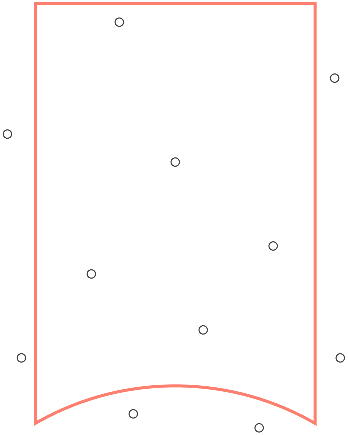
Counting Points



The following is a problem that arises in proving the valence formula for modular functions.

Modular functions are complex functions that satisfy $f(z+1) = f(z)$ and $f(-1/z) = f(z)$.
We want to count the number of zeroes and poles of f inside the curve γ.

Counting Points

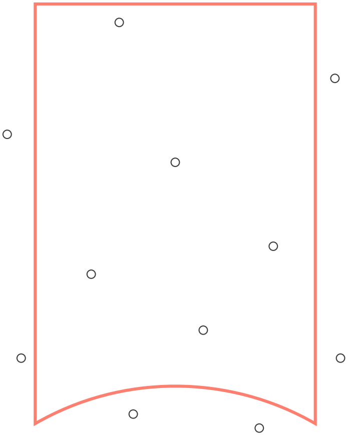


The following is a problem that arises in proving the valence formula for modular functions.

Modular functions are complex functions that satisfy $f(z+1) = f(z)$ and $f(-1/z) = f(z)$.
We want to count the number of zeroes and poles of f inside the curve γ.

This is done by evaluating $\int_\gamma \frac{f'(z)}{f(z)} \, dz$.

Counting Points



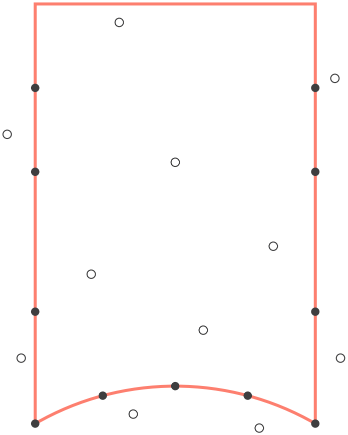
The following is a problem that arises in proving the valence formula for modular functions.

Modular functions are complex functions that satisfy $f(z+1) = f(z)$ and $f(-1/z) = f(z)$.
We want to count the number of zeroes and poles of f inside the curve γ.

This is done by evaluating $\int_\gamma \frac{f'(z)}{f(z)} dz$.

Problem: This integral is not defined if there are zeros/poles directly on γ!

Counting Points



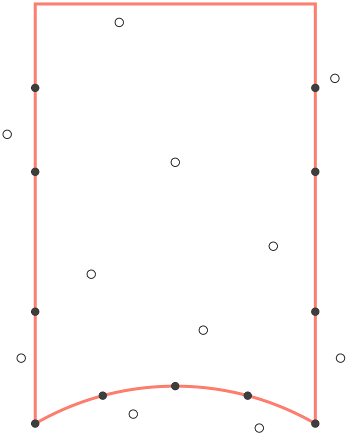
The following is a problem that arises in proving the valence formula for modular functions.

Modular functions are complex functions that satisfy $f(z+1) = f(z)$ and $f(-1/z) = f(z)$.
We want to count the number of zeroes and poles of f inside the curve γ.

This is done by evaluating $\int_\gamma \frac{f'(z)}{f(z)} \mathrm{d}z$.

Problem: This integral is not defined if there are zeros/poles directly on γ!

Counting Points



The following is a problem that arises in proving the valence formula for modular functions.

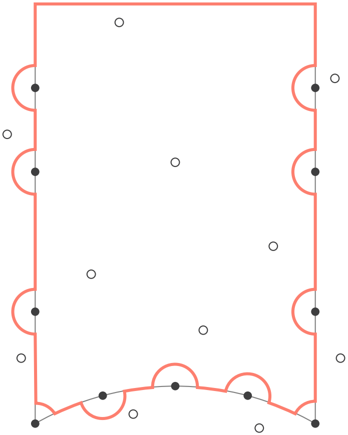
Modular functions are complex functions that satisfy $f(z+1) = f(z)$ and $f(-1/z) = f(z)$.
We want to count the number of zeroes and poles of f inside the curve γ.

This is done by evaluating $\int_\gamma \frac{f'(z)}{f(z)} dz$.

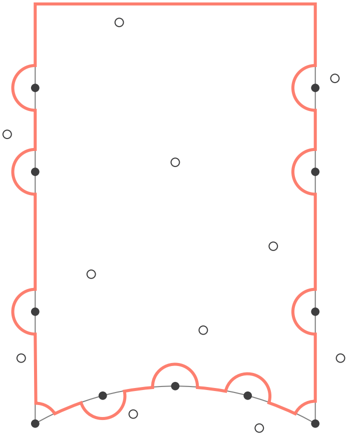
Problem: This integral is not defined if there are zeros/poles directly on γ!

Solution: Deform γ to a contour γ_ε by adding small circular arcs of radius ε, then let $\varepsilon \to 0$.

A Complicated Integration Contour



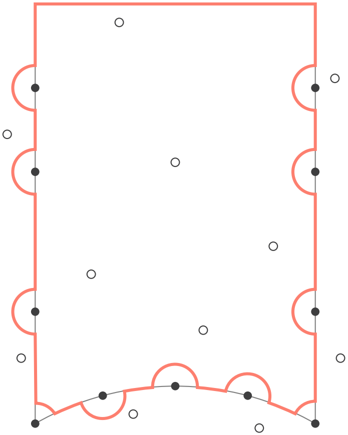
A Complicated Integration Contour



The following geometric facts here are obvious to humans:

- γ_ε is still a simple counter-clockwise closed contour.

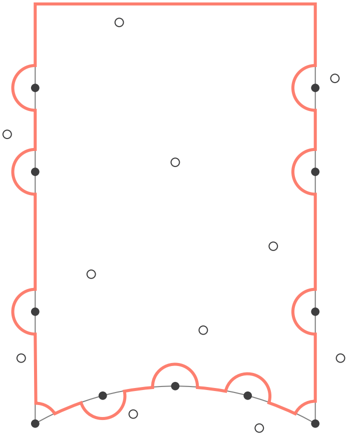
A Complicated Integration Contour



The following geometric facts here are obvious to humans:

- γ_ε is still a simple counter-clockwise closed contour.
- There are no more “bad points” directly on γ_ε.

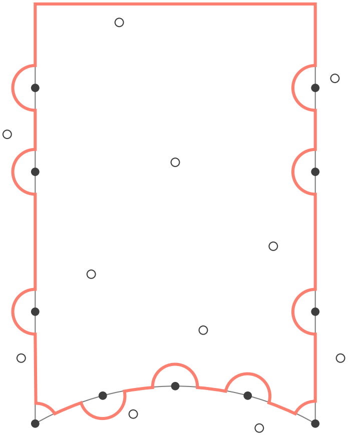
A Complicated Integration Contour



The following geometric facts here are obvious to humans:

- γ_ε is still a simple counter-clockwise closed contour.
- There are no more “bad points” directly on γ_ε.
- Some of the avoided points are inside, others outside (we need to know which).

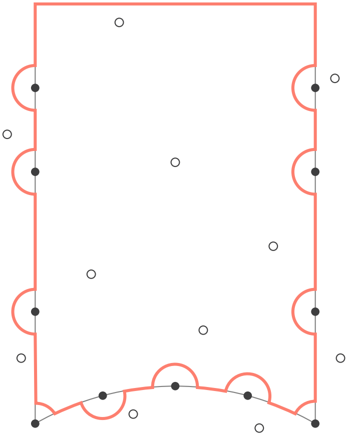
A Complicated Integration Contour



The following geometric facts here are obvious to humans:

- γ_ε is still a simple counter-clockwise closed contour.
- There are no more “bad points” directly on γ_ε.
- Some of the avoided points are inside, others outside (we need to know which).
- All the bad points not directly on the border are unaffected.

A Complicated Integration Contour

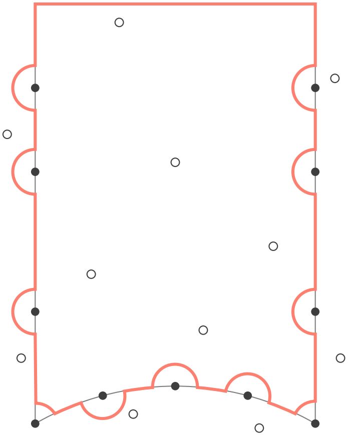


The following geometric facts here are obvious to humans:

- γ_ε is still a simple counter-clockwise closed contour.
- There are no more “bad points” directly on γ_ε.
- Some of the avoided points are inside, others outside (we need to know which).
- All the bad points not directly on the border are unaffected.

A rigorous explicit proof of all of these obvious facts would be so painful it would be unfeasible.

A Complicated Integration Contour



The following geometric facts here are obvious to humans:

- γ_ε is still a simple counter-clockwise closed contour.
- There are no more “bad points” directly on γ_ε.
- Some of the avoided points are inside, others outside (we need to know which).
- All the bad points not directly on the border are unaffected.

A rigorous explicit proof of all of these obvious facts would be so painful it would be unfeasible.

So let’s try something smarter!

The Wiggle Relation

Our intuition tells us that small deformations do not “significantly” change γ.

The Wiggle Relation

Our intuition tells us that small deformations do not "significantly" change γ.

We would like to have some relation $\gamma \approx \gamma_\varepsilon$ that

The Wiggle Relation

Our intuition tells us that small deformations do not “significantly” change γ.

We would like to have some relation $\gamma \approx \gamma_\varepsilon$ that

- captures the notion that γ_ε is equal to γ except for insignificant small local deformations

The Wiggle Relation

Our intuition tells us that small deformations do not “significantly” change γ.

We would like to have some relation $\gamma \approx \gamma_\varepsilon$ that

- captures the notion that γ_ε is equal to γ except for insignificant small local deformations
- is *compositional*, i.e. we can prove $\approx$ for a complicated contour by proving it locally for its constituent contours

The Wiggle Relation

Our intuition tells us that small deformations do not “significantly” change γ.

We would like to have some relation $\gamma \approx \gamma_\varepsilon$ that

- captures the notion that γ_ε is equal to γ except for insignificant small local deformations
- is *compositional*, i.e. we can prove $\approx$ for a complicated contour by proving it locally for its constituent contours
- all relevant properties of γ transfer to γ_ε if we have $\gamma \approx \gamma_\varepsilon$

The Wiggle Relation

Define a relation $\gamma \underset{I/X}{\approx} \gamma_\varepsilon$ where

- γ is a simple closed contour and $(\gamma_\varepsilon)_{\varepsilon>0}$ is a family of paths

The Wiggle Relation

Define a relation $\gamma \underset{I/X}{\approx} \gamma_\varepsilon$ where

- γ is a simple closed contour and $(\gamma_\varepsilon)_{\varepsilon>0}$ is a family of paths
- I and X are sets of points on γ to avoid

The Wiggle Relation

Define a relation $\gamma \underset{I/X}{\approx} \gamma_\varepsilon$ where

- γ is a simple closed contour and $(\gamma_\varepsilon)_{\varepsilon>0}$ is a family of paths
- I and X are sets of points on γ to avoid

such that for all suff. small $\varepsilon > 0$:

The Wiggle Relation

Define a relation $\gamma \underset{I/X}{\approx} \gamma_\varepsilon$ where

- γ is a simple closed contour and $(\gamma_\varepsilon)_{\varepsilon>0}$ is a family of paths
- I and X are sets of points on γ to avoid

such that for all suff. small $\varepsilon > 0$:

- γ_ε is a simple closed contour and homotopic to γ

The Wiggle Relation

Define a relation $\gamma \underset{I/X}{\approx} \gamma_\varepsilon$ where

- γ is a simple closed contour and $(\gamma_\varepsilon)_{\varepsilon>0}$ is a family of paths
- I and X are sets of points on γ to avoid

such that for all suff. small $\varepsilon > 0$:

- γ_ε is a simple closed contour and homotopic to γ
- γ_ε only differs from γ within a ε-neighbourhood of $I \cup X$

The Wiggle Relation

Define a relation $\gamma \underset{I/X}{\approx} \gamma_\varepsilon$ where

- γ is a simple closed contour and $(\gamma_\varepsilon)_{\varepsilon>0}$ is a family of paths
- I and X are sets of points on γ to avoid

such that for all suff. small $\varepsilon > 0$:

- γ_ε is a simple closed contour and homotopic to γ
- γ_ε only differs from γ within a ε-neighbourhood of $I \cup X$
- all points in I are inside γ_ε and all points in X outside

The Wiggle Relation

Define a relation $\gamma \underset{I/X}{\approx} \gamma_\varepsilon$ where

- γ is a simple closed contour and $(\gamma_\varepsilon)_{\varepsilon>0}$ is a family of paths
- I and X are sets of points on γ to avoid

such that for all suff. small $\varepsilon > 0$:

- γ_ε is a simple closed contour and homotopic to γ
- γ_ε only differs from γ within a ε-neighbourhood of $I \cup X$
- all points in I are inside γ_ε and all points in X outside

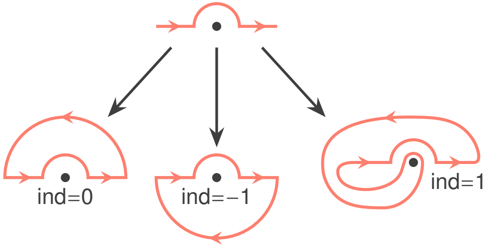
But: For compositionality, we need to generalise to non-closed contours.

The Generalised Wiggle Relation

Problem: The words “inside” and “outside” only make sense for closed contours.

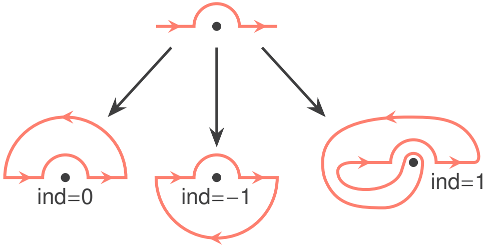
The Generalised Wiggle Relation

Problem: The words “inside” and “outside” only make sense for closed contours.



The Generalised Wiggle Relation

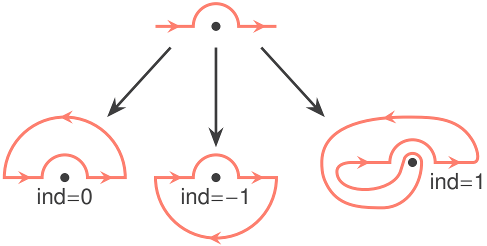
Problem: The words “inside” and “outside” only make sense for closed contours.



Solution: Don’t ask whether the bad point lies inside γ or outside!

The Generalised Wiggle Relation

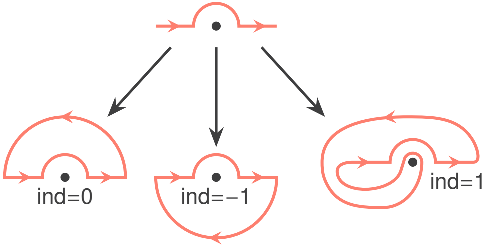
Problem: The words “inside” and “outside” only make sense for closed contours.



Solution: Don’t ask whether the bad point lies inside γ or outside!
Instead ask whether γ avoids the bad point by swerving left  or right .

The Generalised Wiggle Relation

Problem: The words “inside” and “outside” only make sense for closed contours.



Solution: Don’t ask whether the bad point lies inside γ or outside!
Instead ask whether γ avoids the bad point by swerving left  or right .

If γ is a loop, we can easily recover the inside/outside information from this:
For a counter-clockwise loop, left = inside and right = outside.

The Generalised Wiggle Relation

Okay, so we want a relation $\gamma \underset{L/R}{\approx} \gamma_\varepsilon$ that says that γ_ε avoids all points in L by swerving left and those in R by swerving right

The Generalised Wiggle Relation

Okay, so we want a relation $\gamma \underset{L/R}{\approx} \gamma_\varepsilon$ that says that γ_ε avoids all points in L by swerving left  and those in R by swerving right 

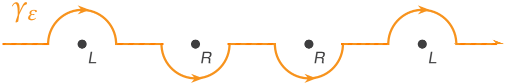
Problem: How do we formally define "γ avoids z by swerving left"?

The Generalised Wiggle Relation

Okay, so we want a relation $\gamma \underset{L/R}{\approx} \gamma_\varepsilon$ that says that γ_ε avoids all points in L by swerving left and those in R by swerving right

Problem: How do we formally define "γ avoids z by swerving left"?

Solution: Additionally keep track of a *dual contour* $\widehat{\gamma}_\varepsilon$ that does the opposite of γ_ε!

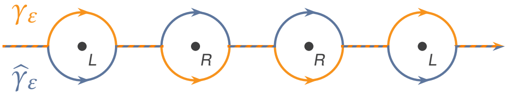


The Generalised Wiggle Relation

Okay, so we want a relation $\gamma \underset{L/R}{\approx} \gamma_\varepsilon$ that says that γ_ε avoids all points in L by swerving left  and those in R by swerving right 

Problem: How do we formally define "γ avoids z by swerving left"?

Solution: Additionally keep track of a *dual contour* $\widehat{\gamma}_\varepsilon$ that does the opposite of γ_ε!

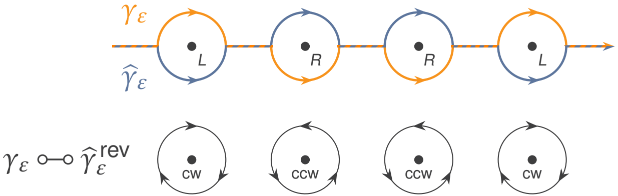


The Generalised Wiggle Relation

Okay, so we want a relation $\gamma \underset{L/R}{\approx} \gamma_\varepsilon$ that says that γ_ε avoids all points in L by swerving left  and those in R by swerving right 

Problem: How do we formally define “γ avoids z by swerving left”?

Solution: Additionally keep track of a *dual contour* $\widehat{\gamma}_\varepsilon$ that does the opposite of γ_ε!

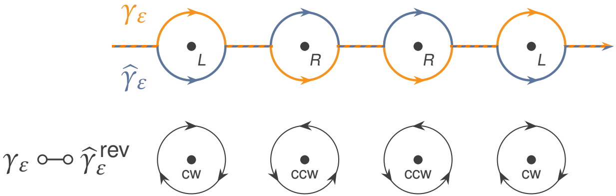


The Generalised Wiggle Relation

Okay, so we want a relation $\gamma \underset{L/R}{\approx} \gamma_\varepsilon$ that says that γ_ε avoids all points in L by swerving left and those in R by swerving right

Problem: How do we formally define "γ avoids z by swerving left"?

Solution: Additionally keep track of a *dual contour* $\widehat{\gamma}_\varepsilon$ that does the opposite of γ_ε!



$$\mathrm{ind}_{\gamma_\varepsilon \circ\!-\!\circ \widehat{\gamma}_\varepsilon^{\mathrm{rev}}}(z) = \begin{cases} -1 & \text{if } z \in L \\ 1 & \text{if } z \in R \end{cases}$$

The Generalised Wiggle Relation

Definition

$\gamma \underset{L/R}{\approx} \gamma_\varepsilon :\Leftrightarrow$ if for all sufficiently small $\varepsilon > 0$:

The Generalised Wiggle Relation

Definition

$\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon$ if for all sufficiently small $\varepsilon > 0$:

- γ_ε and $\widehat{\gamma}_\varepsilon$ are simple contours

The Generalised Wiggle Relation

Definition

$\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon$ if for all sufficiently small $\varepsilon > 0$:

- γ_ε and $\widehat{\gamma}_\varepsilon$ are simple contours
- γ_ε and $\widehat{\gamma}_\varepsilon$ are homotopic to γ within an ε-neighbourhood of $L \cup R$

The Generalised Wiggle Relation

Definition

$\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon$ if for all sufficiently small $\varepsilon > 0$:

- γ_ε and $\widehat{\gamma}_\varepsilon$ are simple contours
- γ_ε and $\widehat{\gamma}_\varepsilon$ are homotopic to γ within an ε-neighbourhood of $L \cup R$
- $\mathrm{ind}_{\gamma_\varepsilon \circ\!\!-\!\!\circ \widehat{\gamma}_\varepsilon^{\mathrm{rev}}} = \mathrm{ind}_{\gamma_\varepsilon}(z) - \mathrm{ind}_{\widehat{\gamma}_\varepsilon}(z) = \begin{cases} -1 & \text{if } z \in L \\ 1 & \text{if } z \in R \end{cases}$

The Generalised Wiggle Relation

Definition

$\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon$ if for all sufficiently small $\varepsilon > 0$:

- γ_ε and $\widehat{\gamma}_\varepsilon$ are simple contours
- γ_ε and $\widehat{\gamma}_\varepsilon$ are homotopic to γ within an ε-neighbourhood of $L \cup R$
- $\mathrm{ind}_{\gamma_\varepsilon \circ\!\!-\!\!\circ \widehat{\gamma}_\varepsilon^{\mathrm{rev}}} = \mathrm{ind}_{\gamma_\varepsilon}(z) - \mathrm{ind}_{\widehat{\gamma}_\varepsilon}(z) = \begin{cases} -1 & \text{if } z \in L \\ 1 & \text{if } z \in R \end{cases}$

Theorem

If $\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon$ and γ is closed, then $\gamma \underset{L/R}{\approx} \gamma_\varepsilon$ if γ is clockwise and $\gamma \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon$ if γ is counter-clockwise.

The Generalised Wiggle Relation

Our new ternary $\approx$ relation is compositional:

$$\frac{}{\gamma \underset{\emptyset/\emptyset}{\approx} \gamma : \gamma} \textsf{REFL}$$

The Generalised Wiggle Relation

Our new ternary $\approx$ relation is compositional:

$$\dfrac{}{\gamma \underset{\emptyset/\emptyset}{\approx} \gamma : \gamma}\ \textsf{REFL} \qquad\qquad \dfrac{\gamma \underset{R/L}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{\gamma \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon : \gamma_\varepsilon}\ \textsf{FLIP}$$

The Generalised Wiggle Relation

Our new ternary $\approx$ relation is compositional:

$$\frac{}{\gamma \underset{\emptyset/\emptyset}{\approx} \gamma : \gamma} \textsf{ REFL}$$

$$\frac{\gamma \underset{R/L}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{\gamma \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon : \gamma_\varepsilon} \textsf{ FLIP}$$

$$\frac{\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{\gamma^{\mathsf{rev}} \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon^{\mathsf{rev}} : \gamma_\varepsilon^{\mathsf{rev}}} \textsf{ REVERSE}$$

The Generalised Wiggle Relation

Our new ternary $\approx$ relation is compositional:

$$\frac{}{\gamma \underset{\emptyset/\emptyset}{\approx} \gamma : \gamma}\ \textsf{REFL} \qquad \frac{\gamma \underset{R/L}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{\gamma \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon : \gamma_\varepsilon}\ \textsf{FLIP} \qquad \frac{\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{\gamma^{\text{rev}} \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon^{\text{rev}} : \gamma_\varepsilon^{\text{rev}}}\ \textsf{REVERSE}$$

$$\frac{\text{end}(\gamma) = \text{start}(\eta) \qquad \gamma \underset{L_1/R_1}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon \qquad \eta \underset{L_2/R_2}{\approx} \eta_\varepsilon : \widehat{\eta}_\varepsilon}{\gamma \multimap \eta \underset{L_1 \cup L_2/R_1 \cup R_2}{\approx} \gamma_\varepsilon \multimap \eta_\varepsilon : \widehat{\gamma}_\varepsilon \multimap \widehat{\eta}_\varepsilon}\ \textsc{Join}$$

The Generalised Wiggle Relation

Our new ternary $\approx$ relation is compositional:

$$\frac{}{\gamma \underset{\emptyset/\emptyset}{\approx} \gamma : \gamma}\ \textsf{REFL} \qquad \frac{\gamma \underset{R/L}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{\gamma \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon : \gamma_\varepsilon}\ \textsf{FLIP} \qquad \frac{\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{\gamma^{\mathsf{rev}} \underset{L/R}{\approx} \widehat{\gamma}_\varepsilon^{\mathsf{rev}} : \gamma_\varepsilon^{\mathsf{rev}}}\ \textsf{REVERSE}$$

$$\frac{\mathsf{end}(\gamma) = \mathsf{start}(\eta) \qquad \gamma \underset{L_1/R_1}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon \qquad \eta \underset{L_2/R_2}{\approx} \eta_\varepsilon : \widehat{\eta}_\varepsilon}{\gamma \multimap\!\!\circ\, \eta \underset{L_1 \cup L_2/R_1 \cup R_2}{\approx} \gamma_\varepsilon \multimap\!\!\circ\, \eta_\varepsilon : \widehat{\gamma}_\varepsilon \multimap\!\!\circ\, \widehat{\eta}_\varepsilon}\ \textsc{Join}$$

This reduces proving $\approx$ for a concrete complicated path to proving it locally for its simple constituent paths!

Basic Avoidance Patterns

Straight line: 

Basic Avoidance Patterns

Straight line: 

Circular arc: 

Basic Avoidance Patterns

Straight line:

Circular arc:

Straight line corner:

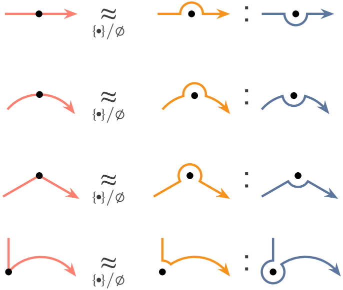
Basic Avoidance Patterns

Straight line:

Circular arc:

Straight line corner:

Line/arc corner:



Basic Avoidance Patterns

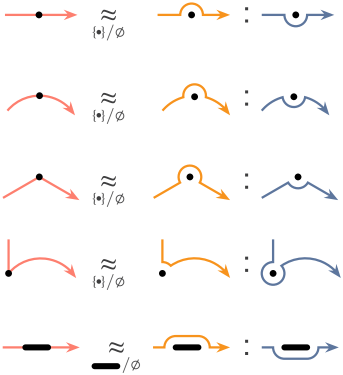
Straight line:

Circular arc:

Straight line corner:

Line/arc corner:

Non-isolated:



Basic Avoidance Patterns

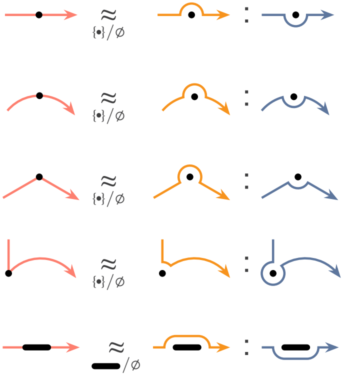
Straight line:

Circular arc:

Straight line corner:

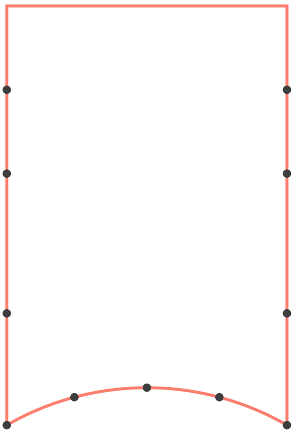
Line/arc corner:

Non-isolated:

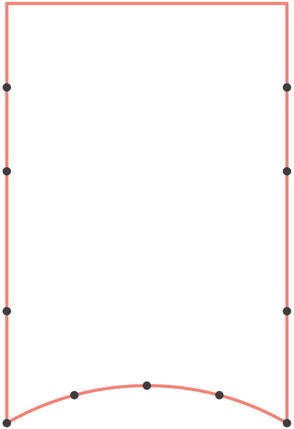


These rules have only trivial side conditions and are thus very easy to apply.

Back to our example

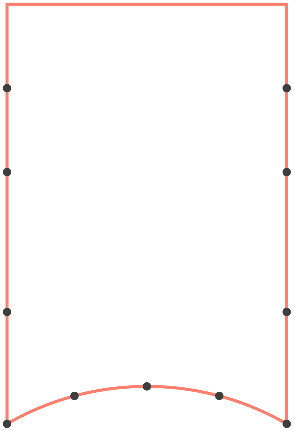


Back to our example



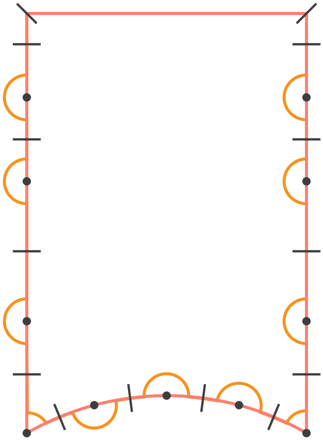
1. Prove that original contour γ is a simple counter-clockwise loop

Back to our example



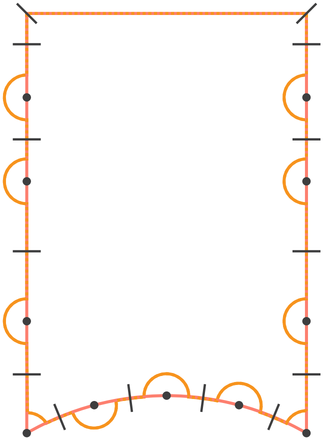
1. Prove that original contour γ is a simple counter-clockwise loop
2. Cut γ at arbitrary point between any pair of adjacent bad points

Back to our example



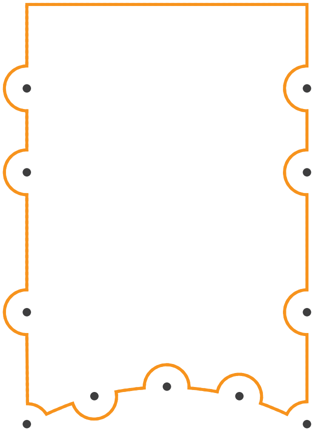
1. Prove that original contour γ is a simple counter-clockwise loop
2. Cut γ at arbitrary point between any pair of adjacent bad points

Back to our example



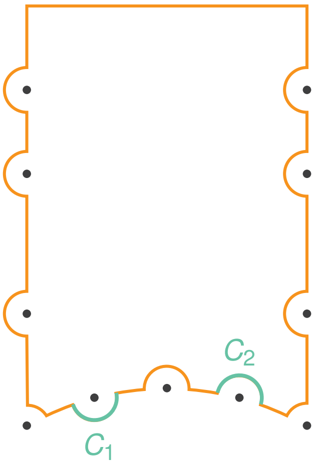
1. Prove that original contour γ is a simple counter-clockwise loop
2. Cut γ at arbitrary point between any pair of adjacent bad points
3. Prove $\approx$ relation for each part using the basic rules

Back to our example



1. Prove that original contour γ is a simple counter-clockwise loop
2. Cut γ at arbitrary point between any pair of adjacent bad points
3. Prove $\approx$ relation for each part using the basic rules
4. Put everything together using JOIN/REFL/FLIP rules

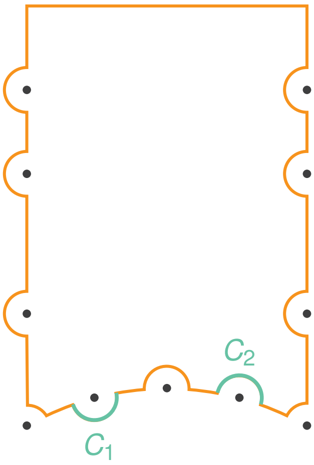
One last complication



One trick Apostol uses to evaluate the integral:

- C_2 is the image of C_1 under $z \mapsto -1/z$.

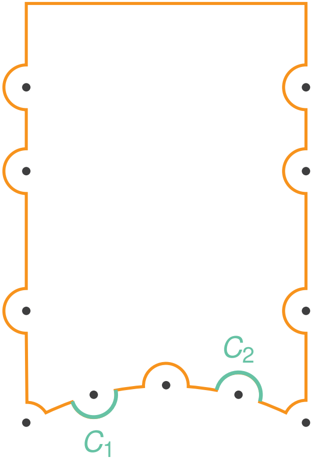
One last complication



One trick Apostol uses to evaluate the integral:

- C_2 is the image of C_1 under $z \mapsto -1/z$.
- Since $f(-1/z) = f(z)$, the integrals $\int C_1$ and $\int C_2$ cancel.

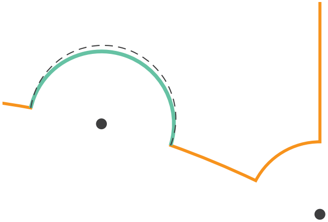
One last complication



One trick Apostol uses to evaluate the integral:

- C_2 is the image of C_1 under $z \mapsto -1/z$.
- Since $f(-1/z) = f(z)$, the integrals $\int C_1$ and $\int C_2$ cancel.
- But: Our C_2 is not actually the image of C_1!

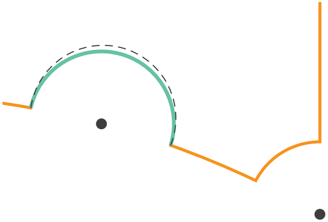
One last complication



One trick Apostol uses to evaluate the integral:

- C_2 is the image of C_1 under $z \mapsto -1/z$.
- Since $f(-1/z) = f(z)$, the integrals $\int C_1$ and $\int C_2$ cancel.
- But: Our C_2 is not actually the image of C_1!

One last complication

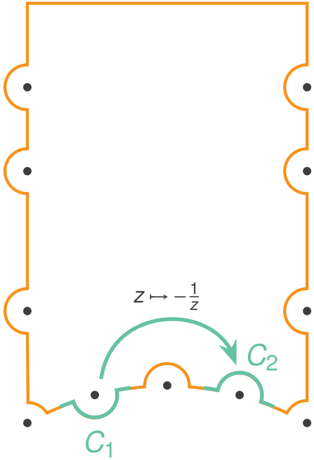


One trick Apostol uses to evaluate the integral:

- C_2 is the image of C_1 under $z \mapsto -1/z$.
- Since $f(-1/z) = f(z)$, the integrals $\int C_1$ and $\int C_2$ cancel.
- But: Our C_2 is not actually the image of C_1!

We thus need to modify our deformed contour to make the argument work.

One last complication



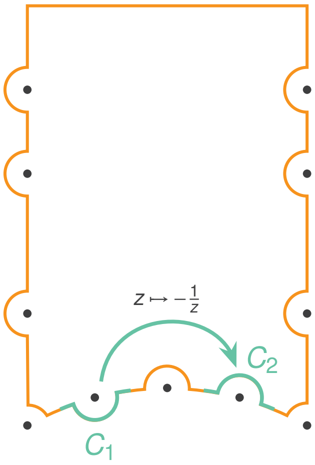
One trick Apostol uses to evaluate the integral:

- C_2 is the image of C_1 under $z \mapsto -1/z$.
- Since $f(-1/z) = f(z)$, the integrals $\int C_1$ and $\int C_2$ cancel.
- But: Our C_2 is not actually the image of C_1!

We thus need to modify our deformed contour to make the argument work.

- C_1 is a circular arc around the bad point
- C_2 is *not* a circular arc around the bad point but rather the image of C_1 under $z \mapsto -\frac{1}{z}$

One last complication



One trick Apostol uses to evaluate the integral:

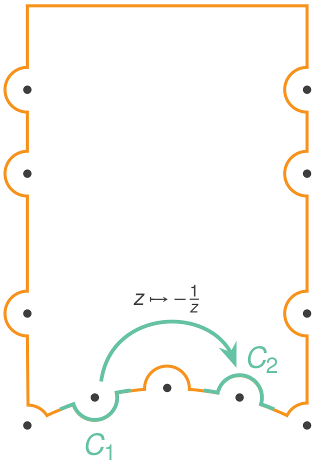
- C_2 is the image of C_1 under $z \mapsto -1/z$.
- Since $f(-1/z) = f(z)$, the integrals $\int C_1$ and $\int C_2$ cancel.
- But: Our C_2 is not actually the image of C_1!

We thus need to modify our deformed contour to make the argument work.

- C_1 is a circular arc around the bad point
- C_2 is *not* a circular arc around the bad point but rather the image of C_1 under $z \mapsto -\frac{1}{z}$

Does the $\approx$ relation still hold?

One last complication



One trick Apostol uses to evaluate the integral:

- C_2 is the image of C_1 under $z \mapsto -1/z$.
- Since $f(-1/z) = f(z)$, the integrals $\int C_1$ and $\int C_2$ cancel.
- But: Our C_2 is not actually the image of C_1!

We thus need to modify our deformed contour to make the argument work.

- C_1 is a circular arc around the bad point
- C_2 is *not* a circular arc around the bad point but rather the image of C_1 under $z \mapsto -\frac{1}{z}$

Does the $\approx$ relation still hold?
Yes, because $\approx$ is closed under nice functions.

Composition

Theorem

Let $\operatorname{img}(\gamma) \subseteq A$ *and* $h : A \to \mathbb{C}$ *be a continuous function of degree 1.*

Composition

Theorem

Let $\operatorname{img}(\gamma) \subseteq A$ *and* $h : A \to \mathbb{C}$ *be a continuous function of degree 1.*

Then:

$$\frac{\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{h \circ \gamma \underset{h(L)/h(R)}{\approx} h \circ \gamma_\varepsilon : h \circ \widehat{\gamma}_\varepsilon} \text{ IMAGE}$$

Composition

Theorem

Let $\operatorname{img}(\gamma) \subseteq A$ *and* $h : A \to \mathbb{C}$ *be a continuous function of degree 1.*

Then:

$$\frac{\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{h \circ \gamma \underset{h(L)/h(R)}{\approx} h \circ \gamma_\varepsilon : h \circ \widehat{\gamma}_\varepsilon} \text{ IMAGE}$$

For degree -1 *one needs to additionally swap left and right.*

Composition

Theorem

Let $\mathrm{img}(\gamma) \subseteq A$ *and* $h : A \to \mathbb{C}$ *be a continuous function of degree 1.*

Then:

$$\frac{\gamma \underset{L/R}{\approx} \gamma_\varepsilon : \widehat{\gamma}_\varepsilon}{h \circ \gamma \underset{h(L)/h(R)}{\approx} h \circ \gamma_\varepsilon : h \circ \widehat{\gamma}_\varepsilon} \text{ IMAGE}$$

For degree -1 *one needs to additionally swap left and right.*

In particular, this works for all affine injective functions and all holomorphic injective functions.

Conclusion

- Formalisation makes some things that are easy on paper quite difficult.

Conclusion

- Formalisation makes some things that are easy on paper quite difficult.
- This forces us to come up with new tricks to get around these difficulties.

Conclusion

- Formalisation makes some things that are easy on paper quite difficult.
- This forces us to come up with new tricks to get around these difficulties.
- It can be useful to look at what other mathematical software does ($\rightarrow$ multiseries)

Conclusion

- Formalisation makes some things that are easy on paper quite difficult.
- This forces us to come up with new tricks to get around these difficulties.
- It can be useful to look at what other mathematical software does ($\rightarrow$ multiseries)
- ... or two try to somehow translate human intuition into a rigorous justification ($\rightarrow$ the “wiggle relation”)

Conclusion

- Formalisation makes some things that are easy on paper quite difficult.
- This forces us to come up with new tricks to get around these difficulties.
- It can be useful to look at what other mathematical software does ($\rightarrow$ multiseries)
- ...or two try to somehow translate human intuition into a rigorous justification ($\rightarrow$ the "wiggle relation")

In any case, thanks to our tools and libraries we are now at a point where a small team can routinely tackle graduate-level maths "for fun".