Semi-Automatic Real Asymptotics in Isabelle/HOL

Manuel Eberl

Technische Universität München

6 June 2018

- Interactive theorem prover; mostly *Higher Order Logic*

- Interactive theorem prover; mostly *Higher Order Logic*
- All theorems are broken down to basic logical inference

- Interactive theorem prover; mostly *Higher Order Logic*
- All theorems are broken down to basic logical inference
- Unlike Coq: No dependent types

- Interactive theorem prover; mostly *Higher Order Logic*
- All theorems are broken down to basic logical inference
- Unlike Coq: No dependent types
- Large library of real and complex analysis

- Interactive theorem prover; mostly *Higher Order Logic*
- All theorems are broken down to basic logical inference
- Unlike Coq: No dependent types
- Large library of real and complex analysis
- Archive of Formal Proofs:
  Large collection of Isabelle proof developments

Problem: Asymptotics in Isabelle are ugly to prove!

Problem: Asymptotics in Isabelle are ugly to prove!

Even trivial things like $x^2 - x \longrightarrow \infty$ require some thinking and several lines of proofs

Problem: Asymptotics in Isabelle are ugly to prove!

Even trivial things like $x^2 - x \longrightarrow \infty$ require some thinking and several lines of proofs

Example: Lemma required for Akra–Bazzi

$$\lim_{x\to\infty} \left(1 - \frac{1}{b\log^{1+\varepsilon} x}\right)^p \left(1 + \frac{1}{\log^{\varepsilon/2}\left(bx + \frac{x}{\log^{1+\varepsilon} x}\right)}\right) - \left(1 + \frac{1}{\log^{\varepsilon/2} x}\right) \;=\; 0^+$$

Problem: Asymptotics in Isabelle are ugly to prove!

Even trivial things like $x^2 - x \longrightarrow \infty$ require some thinking and several lines of proofs

Example: Lemma required for Akra–Bazzi

$$\lim_{x\to\infty} \left(1 - \frac{1}{b\log^{1+\varepsilon} x}\right)^p \left(1 + \frac{1}{\log^{\varepsilon/2}\left(bx + \frac{x}{\log^{1+\varepsilon} x}\right)}\right) - \left(1 + \frac{1}{\log^{\varepsilon/2} x}\right) \;=\; 0^+$$

Original author: ‘Trivial, just Taylor-expand it!’

Problem: Asymptotics in Isabelle are ugly to prove!

Even trivial things like $x^2 - x \longrightarrow \infty$ require some thinking and several lines of proofs

Example: Lemma required for Akra–Bazzi

$$\lim_{x\to\infty} \left(1 - \frac{1}{b\log^{1+\varepsilon} x}\right)^p \left(1 + \frac{1}{\log^{\varepsilon/2}\left(bx + \frac{x}{\log^{1+\varepsilon} x}\right)}\right) - \left(1 + \frac{1}{\log^{\varepsilon/2} x}\right) = 0^+$$

Original author: 'Trivial, just Taylor-expand it!'

In Isabelle: 700 lines of messy proofs

lemma akra_bazzi_aux:
filterlim
$(\lambda x.\ (1 - 1/(b * \ln x\ \hat{}\ (1 + \varepsilon))\ \hat{}\ p) *$
$(1 + \ln\ (b * x + x/\ln x\ \hat{}\ (1 + \varepsilon))\ \hat{}\ (-\varepsilon/2)) -$
$(1 + \ln x\ \hat{}\ (-\varepsilon/2)))$
(at_right 0) at_top

lemma akra_bazzi_aux:
filterlim

$$(\lambda x.\ (1 - 1/(b * \ln x \,\hat{}\, (1+\varepsilon)) \,\hat{}\, p) * (1 + \ln (b * x + x/\ln x \,\hat{}\, (1+\varepsilon)) \,\hat{}\, (-\varepsilon/2)) - (1 + \ln x \,\hat{}\, (-\varepsilon/2)))$$

(at_right 0) at_top

by magic

lemma akra_bazzi_aux:
filterlim
$(\lambda x.\ (1 - 1/(b * \ln x \,\hat{}\,(1+\varepsilon))\,\hat{}\,p) *$
$(1 + \ln (b * x + x/\ln x \,\hat{}\,(1+\varepsilon))\,\hat{}\,(-\varepsilon/2)) -$
$(1 + \ln x \,\hat{}\,(-\varepsilon/2)))$
(at_right 0) at_top
by magic

This is what we would like to have.

lemma akra_bazzi_aux:
filterlim
$(\lambda x.\ (1 - 1/(b * \ln x\,\hat{}\,(1+\varepsilon))\,\hat{}\,p) *$
$(1 + \ln (b * x + x/\ln x\,\hat{}\,(1+\varepsilon))\,\hat{}\,(-\varepsilon/2)) -$
$(1 + \ln x\,\hat{}\,(-\varepsilon/2)))$
(at_right 0) at_top
by magic

This is what we would like to have.

Computer Algebra Systems can do this (sort of)

So why can't we?

Multiseries Expansions

Multiseries Expansions

Disclaimer:

- None of this was invented by me.

Multiseries Expansions

Disclaimer:

- None of this was invented by me.
- I will not show actual Isabelle code for better readability.

Multiseries Expansions

Disclaimer:

- None of this was invented by me.
- I will not show actual Isabelle code for better readability.

Related Work:

- *Asymptotic Expansions of exp–log Functions*
  by Richardson, Salvy, Shackell, van der Hoeven
- *On Computing Limits in a Symbolic Manipulation System*
  by Gruntz

Multiseries Expansions

Power series expansions are insufficient for many important functions: $\exp(x)$, $\ln(x)$, $\Gamma(x)$ for $x \to \infty$

Multiseries Expansions

Power series expansions are insufficient for many important functions: $\exp(x)$, $\ln(x)$, $\Gamma(x)$ for $x \to \infty$

Example:

$$(x + \ln(x))^{-1} \sim \frac{1}{2}x^{-1} - \frac{1}{4}x^{-2}\ln(x) + \frac{1}{8}x^{-3}\ln(x)^2 + \dots$$

Solution: Multiseries

- Like an asymptotic power series, but may contain powers of several 'basis functions' $b_1(x), \dots, b_n(x)$

Multiseries Expansions

Power series expansions are insufficient for many important functions: $\exp(x)$, $\ln(x)$, $\Gamma(x)$ for $x \to \infty$

Example:

$$(x + \ln(x))^{-1} \ \sim \ \tfrac{1}{2}x^{-1} - \tfrac{1}{4}x^{-2}\ln(x) + \tfrac{1}{8}x^{-3}\ln(x)^2 + \dots$$

Solution: Multiseries

- Like an asymptotic power series, but may contain powers of several 'basis functions' $b_1(x), \dots, b_n(x)$
- Formally: $\mathbb{R}[B_1, \dots, B_n]$ or $\mathbb{R}[B_n] \dots [B_1]$

Multiseries Expansions

Power series expansions are insufficient for many important functions: $\exp(x)$, $\ln(x)$, $\Gamma(x)$ for $x \to \infty$

Example:

$$(x + \ln(x))^{-1} \sim \frac{1}{2}x^{-1} - \frac{1}{4}x^{-2}\ln(x) + \frac{1}{8}x^{-3}\ln(x)^2 + \ldots$$

Solution: Multiseries

- Like an asymptotic power series, but may contain powers of several 'basis functions' $b_1(x), \ldots, b_n(x)$
- Formally: $\mathbb{R}[B_1, \ldots, B_n]$ or $\mathbb{R}[B_n] \ldots [B_1]$
- The basis must be ordered descendingly by 'growth class': $\forall i.\ \ln b_{i+1}(x) \in o(\ln b_i(x))$

Multiseries Expansions

Power series expansions are insufficient for many important functions: $\exp(x)$, $\ln(x)$, $\Gamma(x)$ for $x \to \infty$

Example:

$$(x + \ln(x))^{-1} \sim \frac{1}{2}x^{-1} - \frac{1}{4}x^{-2}\ln(x) + \frac{1}{8}x^{-3}\ln(x)^2 + \dots$$

Solution: Multiseries

- Like an asymptotic power series, but may contain powers of several 'basis functions' $b_1(x), \dots, b_n(x)$
- Formally: $\mathbb{R}[B_1, \dots, B_n]$ or $\mathbb{R}[B_n] \dots [B_1]$
- The basis must be ordered descendingly by 'growth class': $\forall i.\ \ln b_{i+1}(x) \in o(\ln b_i(x))$
- Typical basis: $\exp(x^2)$, $\exp(x)$, x, $\ln x$, $\ln \ln x$

A coalgebraic view of Multiseries

type Basis $=$ $(\mathbb{R} \to \mathbb{R})$ list

A coalgebraic view of Multiseries

$$\begin{array}{l}
\textbf{type}\ \mathsf{Basis}\ =\ (\mathbb{R} \to \mathbb{R})\ \mathsf{list} \\
\textbf{datatype}\ \mathsf{MS}\ :\ \mathsf{Basis} \to \mathsf{Type}\ \ \textbf{where} \\
\quad \mathsf{Const}\ :\ \mathbb{R} \to \mathsf{MS}\ [] \\
\quad \mathsf{Series}\ :\ \mathsf{LList}\ (\mathsf{MS}\ bs \times \mathbb{R}) \to \mathsf{MS}\ (b :: bs)
\end{array}$$

A coalgebraic view of Multiseries

$$
\begin{array}{l}
\textbf{type}\ \mathsf{Basis}\ =\ (\mathbb{R} \to \mathbb{R})\ \mathsf{list} \\
\textbf{datatype}\ \mathsf{MS}\ :\ \mathsf{Basis} \to \mathsf{Type}\ \ \textbf{where} \\
\quad \mathsf{Const}\ :\ \mathbb{R} \to \mathsf{MS}\ [] \\
\quad \mathsf{Series}\ :\ \mathsf{LList}\ (\mathsf{MS}\ bs \times \mathbb{R}) \to \mathsf{MS}\ (b :: bs)
\end{array}
$$

Additionally: bases and series must be ‘sorted’.

A coalgebraic view of Multiseries

$$\begin{array}{l} \textbf{type}\ \mathsf{Basis}\ =\ (\mathbb{R} \to \mathbb{R})\ \mathsf{list} \\ \textbf{datatype}\ \mathsf{MS}\ :\ \mathsf{Basis} \to \mathsf{Type}\ \ \textbf{where} \\ \quad \mathsf{Const}\ :\ \mathbb{R} \to \mathsf{MS}\ [] \\ \quad \mathsf{Series}\ :\ \mathsf{LList}\ (\mathsf{MS}\ bs \times \mathbb{R}) \to \mathsf{MS}\ (b :: bs) \end{array}$$

Additionally: bases and series must be ‘sorted’.

Example for a simple operation:

$\mathsf{negate}\ :\ \mathsf{MS}\ bs \to \mathsf{MS}\ bs$

A coalgebraic view of Multiseries

$$\textbf{type}\ \mathsf{Basis}\ =\ (\mathbb{R} \to \mathbb{R})\ \mathsf{list}$$

$$\textbf{datatype}\ \mathsf{MS} : \mathsf{Basis} \to \mathsf{Type}\ \ \textbf{where}$$

$$\quad \mathsf{Const} : \mathbb{R} \to \mathsf{MS}\ []$$

$$\quad \mathsf{Series} : \mathsf{LList}\ (\mathsf{MS}\ bs \times \mathbb{R}) \to \mathsf{MS}\ (b :: bs)$$

Additionally: bases and series must be 'sorted'.

Example for a simple operation:

$$\mathsf{negate} : \mathsf{MS}\ bs \to \mathsf{MS}\ bs$$

$$\mathsf{negate}\ (\mathsf{Const}\ c)\ =\ \mathsf{Const}\ (-c)$$

A coalgebraic view of Multiseries

$$
\begin{aligned}
&\textbf{type}\ \mathsf{Basis}\ =\ (\mathbb{R} \to \mathbb{R})\ \mathsf{list} \\
&\textbf{datatype}\ \mathsf{MS} : \mathsf{Basis} \to \mathsf{Type}\ \textbf{where} \\
&\quad \mathsf{Const} : \mathbb{R} \to \mathsf{MS}\ [] \\
&\quad \mathsf{Series} : \mathsf{LList}\ (\mathsf{MS}\ bs \times \mathbb{R}) \to \mathsf{MS}\ (b :: bs)
\end{aligned}
$$

Additionally: bases and series must be 'sorted'.

Example for a simple operation:

$$
\begin{aligned}
&\mathsf{negate} : \mathsf{MS}\ bs \to \mathsf{MS}\ bs \\
&\mathsf{negate}\ (\mathsf{Const}\ c)\ =\ \mathsf{Const}\ (-c) \\
&\mathsf{negate}\ (\mathsf{Series}\ ts)\ =\ \mathsf{Series}\ [(\mathsf{negate}\ c, e) \mid (c, e) \leftarrow ts]
\end{aligned}
$$

More complicated operations

- Basic operations (defined corecursively):
  constants, identity, addition, multiplication

More complicated operations

- Basic operations (defined corecursively):
  constants, identity, addition, multiplication
- Substitution into convergent power series:
  Gives us division; ln, exp, sin, etc. at non-singular points

More complicated operations

- Basic operations (defined corecursively):
  constants, identity, addition, multiplication
- Substitution into convergent power series:
  Gives us division; ln, exp, sin, etc. at non-singular points
- exp and ln at singular points require specialised procedures and may add new basis elements

More complicated operations

- Basic operations (defined corecursively):
  constants, identity, addition, multiplication
- Substitution into convergent power series:
  Gives us division; ln, exp, sin, etc. at non-singular points
- exp and ln at singular points require specialised procedures and may add new basis elements
- For operations like Γ, erf, li:
  factor out singularities and treat them separately

Connecting Series and Functions

For simple power series, $f \sim ts$ can be expressed coinductively:

$$\frac{f(x) \in O(x^e) \qquad f(x) - c\,x^e \;\sim\; ts}{f(x) \;\sim\; (c, e) :: ts}$$

Connecting Series and Functions

For simple power series, $f \sim ts$ can be expressed coinductively:

$$\frac{f(x) \in O(x^e) \qquad f(x) - c\,x^e \;\sim\; ts}{f(x) \;\sim\; (c, e) :: ts}$$

Operations are defined corecursively; correctness is proven coinductively. Both are straightforward.

Connecting Series and Functions

For simple power series, $f \sim ts$ can be expressed coinductively:

$$\frac{f(x) \in O(x^e) \qquad f(x) - c\,x^e \;\sim\; ts}{f(x) \;\sim\; (c, e) :: ts}$$

Operations are defined corecursively; correctness is proven coinductively. Both are straightforward.

(Requires some advanced features like ‘Coinduction up to friends’)

Connecting Series and Functions

For simple power series, $f \sim ts$ can be expressed coinductively:

$$\frac{f(x) \in O(x^e) \qquad f(x) - c\,x^e \;\sim\; ts}{f(x) \;\sim\; (c, e) :: ts}$$

Operations are defined corecursively; correctness is proven coinductively. Both are straightforward.

(Requires some advanced features like 'Coinduction up to friends')

Similar for multiseries.

Finding expansions

We can construct expansions for functions ‘bottom up’:

Finding expansions

We can construct expansions for functions 'bottom up':

Example

Find an expansion for $\sin(1/x) + \exp(x)$ for $x \to \infty$:

Finding expansions

We can construct expansions for functions 'bottom up':

Example

Find an expansion for $\sin(1/x) + \exp(x)$ for $x \to \infty$:

- $1/x$ has the trivial expansion x^{-1}

Finding expansions

We can construct expansions for functions 'bottom up':

Example

Find an expansion for $\sin(1/x) + \exp(x)$ for $x \to \infty$:

- $1/x$ has the trivial expansion x^{-1}
- substitute the series x^{-1} into the Taylor expansion of sin

Finding expansions

We can construct expansions for functions 'bottom up':

Example

Find an expansion for $\sin(1/x) + \exp(x)$ for $x \to \infty$:

- $1/x$ has the trivial expansion x^{-1}
- substitute the series x^{-1} into the Taylor expansion of sin
- $\exp(x)$ has to be added as a new basis element

Finding expansions

We can construct expansions for functions 'bottom up':

Example

Find an expansion for $\sin(1/x) + \exp(x)$ for $x \to \infty$:

- $1/x$ has the trivial expansion x^{-1}
- substitute the series x^{-1} into the Taylor expansion of sin
- $\exp(x)$ has to be added as a new basis element
- $\exp(x)$ then has the trivial expansion $\exp(x)$

Finding expansions

We can construct expansions for functions ‘bottom up’:

Example

Find an expansion for $\sin(1/x) + \exp(x)$ for $x \to \infty$:

- $1/x$ has the trivial expansion x^{-1}
- substitute the series x^{-1} into the Taylor expansion of sin
- $\exp(x)$ has to be added as a new basis element
- $\exp(x)$ then has the trivial expansion $\exp(x)$
- add expansions for $\sin(1/x)$ and $\exp(x)$

Finding expansions

End result: Theorem that $\sin(1/x) + \exp(x)$ has the following expansion w. r. t. basis $(\exp(x), x)$:

Finding expansions

End result: Theorem that $\sin(1/x) + \exp(x)$ has the following expansion w. r. t. basis $(\exp(x), x)$:

$$\text{lift\_expansion}\ (\text{sin\_ms}\ (\text{Series}\ [(1, -1)])) + \text{Series}\ [(\text{Series}\ [(1, 0)], 1)]$$

Finding expansions

End result: Theorem that $\sin(1/x) + \exp(x)$ has the following expansion w. r. t. basis $(\exp(x), x)$:

$$\text{lift\_expansion (sin\_ms (Series } [(1, -1)])) + \text{Series } [(\text{Series } [(1, 0)], 1)]$$

which evaluates to

$$\exp(x) + x^{-1} - \frac{1}{6}x^{-3} + \frac{1}{120}x^{-5} - \ldots$$

The sign determination problem

Problem:

- Many operations involve comparisons of real numbers

The sign determination problem

Problem:

- Many operations involve comparisons of real numbers
- ‘Trimming’ expansions involves zeroness tests of real functions

The sign determination problem

Problem:

- Many operations involve comparisons of real numbers
- 'Trimming' expansions involves zeroness tests of real functions
- Both of these are difficult or even undecidable

The sign determination problem

Solution: Heuristic approach using Isabelle's automation

The sign determination problem

Solution: Heuristic approach using Isabelle's automation

- Use automation to determine signs – might fail

The sign determination problem

Solution: Heuristic approach using Isabelle's automation

- Use automation to determine signs – might fail
- Use automation to determine if function is identically zero – might cause non-termination

The sign determination problem

Solution: Heuristic approach using Isabelle's automation

- Use automation to determine signs – might fail
- Use automation to determine if function is identically zero – might cause non-termination
- Optionally: Use approximation by interval arithmetic

The sign determination problem

Solution: Heuristic approach using Isabelle's automation

- Use automation to determine signs – might fail
- Use automation to determine if function is identically zero – might cause non-termination
- Optionally: Use approximation by interval arithmetic
- User may have to supply additional facts

The sign determination problem

Solution: Heuristic approach using Isabelle's automation

- Use automation to determine signs – might fail
- Use automation to determine if function is identically zero – might cause non-termination
- Optionally: Use approximation by interval arithmetic
- User may have to supply additional facts

This works surprisingly well

Proof method

With some pre-processing, we can automatically prove statements of the form

- $f(x) \longrightarrow c$
- $f(x) \sim g(x)$
- $f(x) \in L(g(x))$ for any Landau symbol L

as $x \to l$ for $l \in \mathbb{R} \cup \{\pm\infty\}$

Proof method

With some pre-processing, we can automatically prove statements of the form

- $f(x) \longrightarrow c$
- $f(x) \sim g(x)$
- $f(x) \in L(g(x))$ for any Landau symbol L

as $x \to l$ for $l \in \mathbb{R} \cup \{\pm\infty\}$

f and g can be built from $+$ $-$ $\cdot$ $/$ ln exp min max ˆ $|\cdot|$ $\sqrt[n]{\cdot}$ without restrictions

Proof method

With some pre-processing, we can automatically prove statements of the form

- $f(x) \longrightarrow c$
- $f(x) \sim g(x)$
- $f(x) \in L(g(x))$ for any Landau symbol L

as $x \to l$ for $l \in \mathbb{R} \cup \{\pm\infty\}$

f and g can be built from $+$ $-$ $\cdot$ $/$ ln exp min max ^ $|\cdot|$ $\sqrt[n]{\cdot}$ without restrictions

sin, cos, tan at finite points also possible.

Proof method

Problem: What about 'oscillating' functions like sin, $\lfloor \cdot \rfloor$, mod?

Proof method

Problem: What about 'oscillating' functions like sin, $\lfloor \cdot \rfloor$, mod?

Obvious solution: Asymptotic interval arithmetic:

Proof method

Problem: What about 'oscillating' functions like sin, $\lfloor \cdot \rfloor$, mod?

Obvious solution: Asymptotic interval arithmetic:

- $\sin x \in [-1; 1]$

Proof method

Problem: What about 'oscillating' functions like sin, $\lfloor\cdot\rfloor$, mod?

Obvious solution: Asymptotic interval arithmetic:

- $\sin x \in [-1; 1]$
- $\lfloor x \rfloor \in [x - 1; x]$

Proof method

Problem: What about 'oscillating' functions like sin, $\lfloor\cdot\rfloor$, mod?

Obvious solution: Asymptotic interval arithmetic:

- $\sin x \in [-1; 1]$
- $\lfloor x\rfloor \in [x-1; x]$

Result: Pair of asymptotic lower/upper bound with known multiseries expansion

Proof method

Problem: What about 'oscillating' functions like sin, $\lfloor\cdot\rfloor$, mod?

Obvious solution: Asymptotic interval arithmetic:

- $\sin x \in [-1; 1]$
- $\lfloor x \rfloor \in [x - 1; x]$

Result: Pair of asymptotic lower/upper bound with known multiseries expansion

Works in many cases, but does not cope well with cancellations.

Proof method

Problem: What about 'oscillating' functions like sin, $\lfloor\cdot\rfloor$, mod?

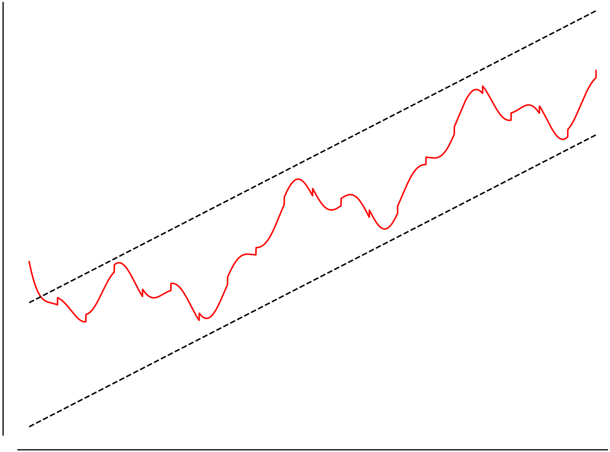
Obvious solution: Asymptotic interval arithmetic:

- $\sin x \in [-1; 1]$
- $\lfloor x \rfloor \in [x - 1; x]$

Result: Pair of asymptotic lower/upper bound with known multiseries expansion

Works in many cases, but does not cope well with cancellations. Good enough.

Proof method



Proof method

Example

lemma $(\lambda n.\ (1 + 1/n)\ \hat{}\ n) \longrightarrow \text{exp}\ 1$
by exp_log_asymptotics

Proof method

Example

lemma $(\lambda n.\ (1 + 1/n)\ \hat{}\ n) \longrightarrow \exp\ 1$
  by exp_log_asymptotics

Example

lemma $(\lambda n.\ (1 + a/n)\ \hat{}\ n) \longrightarrow \exp\ a$
  by exp_log_asymptotics

Proof method

Example

lemma akra_bazzi_aux:
  assumes $b \in \{0 <..< 1\}$ **and** $\varepsilon > 0$
  shows filterlim $(\lambda x.$
    $(1 - H/(b * \ln x \,\hat{}\, (1 + \varepsilon)))\,\hat{}\, p *$
    $(1 + \ln (b * x + H * x/\ln x \,\hat{}\, (1 + \varepsilon))\,\hat{}\,(-\varepsilon/2)) -$
    $(1 + \ln x \,\hat{}\,(-\varepsilon/2)))$
   $(\text{at\_right } 0)$ at_top

Proof method

Example

lemma akra_bazzi_aux:
  assumes $b \in \{0 <..< 1\}$ **and** $\varepsilon > 0$
  shows filterlim $(\lambda x.$
      $(1 - H/(b * \ln x\,\hat{}\,(1+\varepsilon)))\,\hat{}\,p *$
      $(1 + \ln\,(b * x + H * x/\ln x\,\hat{}\,(1+\varepsilon))\,\hat{}\,(-\varepsilon/2)) -$
      $(1 + \ln x\,\hat{}\,(-\varepsilon/2)))$
    (at_right 0) at_top
  by (real_asymp simp: mult_neg_pos)

Comparison to CASs

Similar algorithm by Gruntz in Maple in 1996

Comparison to CASs

Similar algorithm by Gruntz in Maple in 1996
(is now part of Mathematica)

Comparison to CASs

Similar algorithm by Gruntz in Maple in 1996
(is now part of Mathematica)

Back then, all CASs gave wrong results
for many of his test cases!

Comparison to CASs

Similar algorithm by Gruntz in Maple in 1996
(is now part of Mathematica)

Back then, all CASs gave wrong results
for many of his test cases!

Nowadays, most of them work

Comparison to CASs

23 test cases lie in the fragment we support

Comparison to CASs

23 test cases lie in the fragment we support

All of them work automatically

Comparison to CASs

23 test cases lie in the fragment we support

All of them work automatically

Maximum time: 1.726 s; Median: 0.311 s

Comparison to CASs

23 test cases lie in the fragment we support

All of them work automatically

Maximum time: 1.726 s; Median: 0.311 s

Mathematica and Maple do all of them
very quickly and correctly

Comparison to CASs

23 test cases lie in the fragment we support

All of them work automatically

Maximum time: 1.726 s; Median: 0.311 s

Mathematica and Maple do all of them
very quickly and correctly

Maxima and SymPy fail on some of them

Comparison to CASs

23 test cases lie in the fragment we support

All of them work automatically

Maximum time: 1.726 s; Median: 0.311 s

Mathematica and Maple do all of them very quickly and correctly

Maxima and SymPy fail on some of them

Maxima takes very long for some of them and gives wrong result for this:

$$\exp\left(\frac{\log\log\left(x+e^{\log x \log\log x}\right)}{\log\log\log\left(e^x+x+\ln x\right)}\right) \longrightarrow e$$

Comparison to CASs

How well are we doing?

Surprisingly, we are not that much slower (sometimes even faster) than Maple/Mathematica on many examples

Comparison to CASs

How well are we doing?

Surprisingly, we are not that much slower (sometimes even faster) than Maple/Mathematica on many examples

Also: All CASs seem to fail on the Akra–Bazzi example as soon as variables are involved

Comparison to CASs

How well are we doing?

Surprisingly, we are not that much slower (sometimes even faster) than Maple/Mathematica on many examples

Also: All CASs seem to fail on the Akra–Bazzi example as soon as variables are involved

In general, of course, Mathematica/Maple are much better in both scope and speed

Comparison to CASs

How well are we doing?

Surprisingly, we are not that much slower (sometimes even faster) than Maple/Mathematica on many examples

Also: All CASs seem to fail on the Akra–Bazzi example as soon as variables are involved

In general, of course, Mathematica/Maple are much better in both scope and speed

But: you have to trust the implementations.

Comparison to CASs

How well are we doing?

Surprisingly, we are not that much slower (sometimes even faster) than Maple/Mathematica on many examples

Also: All CASs seem to fail on the Akra–Bazzi example as soon as variables are involved

In general, of course, Mathematica/Maple are much better in both scope and speed

But: you have to trust the implementations.

Isabelle still isn't a CAS – but we're getting there.

Questions? Demo?