Calcium: computing in *exact* real and complex fields

Paper: https://arxiv.org/abs/2011.01728

Fredrik Johansson

LFANT seminar November 17, 2020 Bordeaux

Computing in $\mathbb{R}$ and $\mathbb{C}$

- Arithmetic: $x + y$, $x - y$, xy, x/y
- Comparisons and predicates: $x = y$, $x < y$, $x \in \mathbb{Q}, \ldots$
- Number parts: $\operatorname{sgn}(x)$, $|x|$, $\operatorname{Re}(x)$, $\overline{x}$, $\arg(x)$, $\lfloor x \rfloor$, $\ldots$
- Functions and constants: i, π, γ, $\sqrt{x}$, e^x, $\log(x)$, $\zeta(x)$, $\ldots$
- Limits: $\lim_{N \to \infty} f(N)$, $\int_a^b f(x) dx$, $f'(x)$, $\ldots$

"Computable" real number x: there is a program that computes $x_n \in \mathbb{Q}$ with $|x - x_n| < 2^{-n}$

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic

$$(1/49) \cdot 49 \neq 1$$

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic
$$(1/49) \cdot 49 \neq 1$$
- Interval arithmetic, ball arithmetic
$$[3.14 \pm 0.01] - [3.14 \pm 0.01] \neq 0$$

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic

$$(1/49) \cdot 49 \neq 1$$

- Interval arithmetic, ball arithmetic

$$[3.14 \pm 0.01] - [3.14 \pm 0.01] \neq 0$$

- Lazy infinite-precision real numbers

$$3, 3.14, 3.141, 3.1415, 3.14159, \ldots = \pi \text{ ?}$$

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic
$$(1/49) \cdot 49 \neq 1$$
- Interval arithmetic, ball arithmetic
$$[3.14 \pm 0.01] - [3.14 \pm 0.01] \neq 0$$
- Lazy infinite-precision real numbers
$$3, 3.14, 3.141, 3.1415, 3.14159, \ldots = \pi \text{ ?}$$
- Formal rational functions
$$\frac{\sqrt{2}}{2} - \frac{1}{\sqrt{2}} = 0 \quad \text{but} \quad \frac{x}{2} - \frac{1}{x} = \frac{x^2-2}{2x} \neq 0 \text{ in } \mathbb{Q}(x)$$

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic
$$(1/49) \cdot 49 \neq 1$$
- Interval arithmetic, ball arithmetic
$$[3.14 \pm 0.01] - [3.14 \pm 0.01] \neq 0$$
- Lazy infinite-precision real numbers
$$3, 3.14, 3.141, 3.1415, 3.14159, \ldots = \pi \ ?$$
- Formal rational functions
$$\frac{\sqrt{2}}{2} - \frac{1}{\sqrt{2}} = 0 \quad \text{but} \quad \frac{x}{2} - \frac{1}{x} = \frac{x^2-2}{2x} \neq 0 \text{ in } \mathbb{Q}(x)$$
- Abstract number fields, quotient rings
$$\mathbb{Q}[x]/\langle x^2 - 2 \rangle \ \to \ x \cong \sqrt{2} \text{ or } x \cong -\sqrt{2}?$$

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic
$$(1/49) \cdot 49 \neq 1$$
- Interval arithmetic, ball arithmetic
$$[3.14 \pm 0.01] - [3.14 \pm 0.01] \neq 0$$
- Lazy infinite-precision real numbers
$$3, 3.14, 3.141, 3.1415, 3.14159, \ldots = \pi \text{ ?}$$
- Formal rational functions
$$\tfrac{\sqrt{2}}{2} - \tfrac{1}{\sqrt{2}} = 0 \quad \text{but} \quad \tfrac{x}{2} - \tfrac{1}{x} = \tfrac{x^2-2}{2x} \neq 0 \text{ in } \mathbb{Q}(x)$$
- Abstract number fields, quotient rings
$$\mathbb{Q}[x]/\langle x^2 - 2\rangle \;\to\; x \cong \sqrt{2} \text{ or } x \cong -\sqrt{2}?$$
- Embedded number fields, algebraic numbers
$$\pi \notin \overline{\mathbb{Q}}$$

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic

$$(1/49) \cdot 49 \neq 1$$

- Interval arithmetic, ball arithmetic

$$[3.14 \pm 0.01] - [3.14 \pm 0.01] \neq 0$$

- Lazy infinite-precision real numbers

$$3, 3.14, 3.141, 3.1415, 3.14159, \ldots = \pi \text{ ?}$$

- Formal rational functions

$$\frac{\sqrt{2}}{2} - \frac{1}{\sqrt{2}} = 0 \quad \text{but} \quad \frac{x}{2} - \frac{1}{x} = \frac{x^2-2}{2x} \neq 0 \text{ in } \mathbb{Q}(x)$$

- Abstract number fields, quotient rings

$$\mathbb{Q}[x]/\langle x^2 - 2\rangle \;\rightarrow\; x \cong \sqrt{2} \text{ or } x \cong -\sqrt{2}?$$

- Embedded number fields, algebraic numbers

$$\pi \notin \overline{\mathbb{Q}}$$

- Symbolic expressions

A priori meaningless, need concrete semantics/algorithms

Things that are not $\mathbb{R}$ or $\mathbb{C}$

- Floating-point arithmetic

$$(1/49) \cdot 49 \neq 1$$

- Interval arithmetic, ball arithmetic

$$[3.14 \pm 0.01] - [3.14 \pm 0.01] \neq 0$$

- Lazy infinite-precision real numbers

$$3, 3.14, 3.141, 3.1415, 3.14159, \ldots = \pi \ ?$$

- Formal rational functions

$$\frac{\sqrt{2}}{2} - \frac{1}{\sqrt{2}} = 0 \quad \text{but} \quad \frac{x}{2} - \frac{1}{x} = \frac{x^2-2}{2x} \neq 0 \text{ in } \mathbb{Q}(x)$$

- Abstract number fields, quotient rings

$$\mathbb{Q}[x]/\langle x^2 - 2\rangle \ \rightarrow \ x \cong \sqrt{2} \text{ or } x \cong -\sqrt{2}?$$

- Embedded number fields, algebraic numbers

$$\pi \notin \overline{\mathbb{Q}}$$

- Symbolic expressions

A priori meaningless, need concrete semantics/algorithms

...but *what if we put all of that together?*

Idea

- Numbers as field elements $z \in \mathbb{Q}(a_1, \ldots, a_n)$
- Computable *extension numbers* a_k are generated as needed
- Extension numbers are defined symbolically, can be algebraic or transcendental
- Algebraic relations handled using reduction by an ideal

Idea

- Numbers as field elements $z \in \mathbb{Q}(a_1, \ldots, a_n)$
- Computable *extension numbers* a_k are generated as needed
- Extension numbers are defined symbolically, can be algebraic or transcendental
- Algebraic relations handled using reduction by an ideal

Precursors and inspiration

- Symbolic systems (Maple, Mathematica, SymPy, etc.)
  - But more structured, and with stronger semantics
- Magma's `ACF`
  - Not a convenient impl. of $\overline{\mathbb{Q}}$: embedding is random
- Sage's `QQbar`
  - Mixed symbolic expressions / number fields
  - Univariate fields only; severe performance problems
- Elementary fields (Richardson's algorithm)

Calcium

- C library for exact real and complex numbers
- Documentation: `http://fredrikj.net/calcium/`
- LGPL v2.1+ license
- Current version: 0.3-git, 30,000 lines of code
- Includes a Python interface (experimental)
- Dependencies:
  - Flint (polynomial arithmetic, factoring, LLL)
  - Arb (arbitrary-precision ball arithmetic)
  - Antic (number field arithmetic)
  - GMP, MPFR

Some examples

$$\frac{\pi^2 - 9}{\pi + 3} = \pi - 3$$

```
>>> from pyca import *

>>> (pi**2 - 9) / (pi + 3)
0.141593 {a-3 where a = 3.14159 [Pi]}

>>> _ == pi - 3
True
```

Some examples

$$\frac{\varphi^{100} - (1-\varphi)^{100}}{\sqrt{5}} = F_{100}$$

```
>>> phi = (sqrt(5)+1)/2
>>> (phi**100 - (1-phi)**100)/sqrt(5)
3.54225e+20 {354224848179261915075}
```

Some examples

$$\sqrt{5 + 2\sqrt{6}} = \sqrt{2} + \sqrt{3}$$

```
>>> sqrt(5 + 2*sqrt(6))
3.14626 {a where a = 3.14626 [Sqrt(9.89898 {2*b+5})], b =
    2.44949 [b^2-6=0]}
>>> sqrt(2) + sqrt(3)
3.14626 {a+b where a = 1.73205 [a^2-3=0], b = 1.41421 [b
    ^2-2=0]}

>>> sqrt(5 + 2*sqrt(6)) - sqrt(2) - sqrt(3)
0e-1126 {a-c-d where a = 3.14626 [Sqrt(9.89898 {2*b+5})],
    b = 2.44949 [b^2-6=0], c = 1.73205 [c^2-3=0], d =
    1.41421 [d^2-2=0]}
>>> sqrt(5 + 2*sqrt(6)) == sqrt(2) + sqrt(3)
True
```

Some examples

$$4\,\mathrm{atan}\left(\frac{1}{5}\right) - \mathrm{atan}\left(\frac{1}{239}\right) = \frac{\pi}{4}$$

```
>>> 4*atan(ca(1)/5) - atan(ca(1)/239)
0.785398 + 0e-34*I {(a*c-4*b*c)/2 where a = 0e-35 +
    0.00836815*I [Log(0.999965 + 0.00836805*I {(239*c
    +28560)/28561})], b = 0e-34 + 0.394791*I [Log(0.923077
     + 0.384615*I {(5*c+12)/13})], c = I [c^2+1=0]}

>>> pi/4
0.785398 {(a)/4 where a = 3.14159 [Pi]}

>>> 4*atan(ca(1)/5) - atan(ca(1)/239) - pi/4
0
```

Some examples

$$\operatorname{erf}(e^{\pi i/3}) - \operatorname{erfc}(e^{-2\pi i/3}) = -1$$

$$\frac{\Gamma(\pi+1)}{\Gamma(\pi)} = \pi$$

```
>>> erf(exp(pi*i/3)) - erfc(exp(-2*pi*i/3))
-1

>>> gamma(pi+1) / gamma(pi) == pi
True
```

Some examples

$$e^{\pi\sqrt{163}} \neq 640320^3 + 744$$

```
>>> exp(pi*sqrt(163))
2.62537e+17 {a where a = 2.62537e+17 [Exp(40.1092 {b*c})],
     b = 3.14159 [Pi], c = 12.7671 [c^2-163=0]}

>>> ca(640320**3 + 744)
2.62537e+17 {262537412640768744}

>>> exp(pi*sqrt(163)) == (640320**3 + 744)
False

>>> exp(pi*sqrt(163)) - (640320**3 + 744)
-7.49927e-13 {a-262537412640768744 where a = 2.62537e+17 [
    Exp(40.1092 {b*c})], b = 3.14159 [Pi], c = 12.7671 [c
    ^2-163=0]}
```

Some examples

$$i^i = \exp\left(\frac{\pi}{\left(\left(\sqrt{-2}\right)^{\sqrt{2}}\right)^{\sqrt{2}}}\right)$$

```
>>> i**i
0.207880 {a where a = 0.207880 [Pow(1.00000*I {b},
    1.00000*I {b})], b = I [b^2+1=0]}

>>> exp(pi / (sqrt(-2)**sqrt(2))**sqrt(2))
0.207880 {a where a = 0.207880 [Exp(-1.57080 {(-b)/2})],
          b = 3.14159 [Pi]}

>>> i**i - exp(pi / (sqrt(-2)**sqrt(2))**sqrt(2))
0
```

Some examples

$$e^{\log(A)} = A, \quad A = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

```
>>> A = ca_mat([[0,0,1],[0,1,0],[1,0,0]])
>>> B = A.log()

>>> B / (pi * i)
ca_mat of size 3 x 3
[  0.500000 {1/2}, 0, -0.500000 {-1/2}]
[               0, 0,                0]
[-0.500000 {-1/2}, 0, 0.500000 {1/2}]

>>> B.exp()
ca_mat of size 3 x 3
[0, 0, 1]
[0, 1, 0]
[1, 0, 0]
```

But it's not perfect...

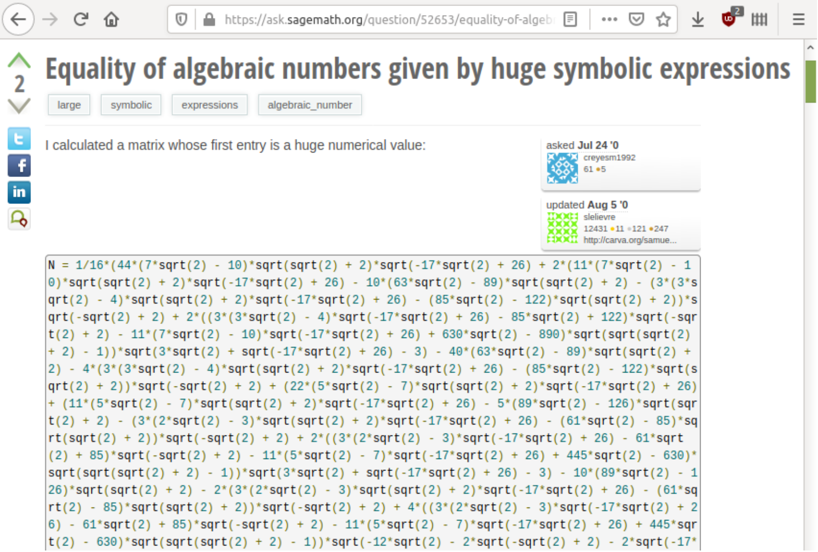
```
>>> A = ca_mat([[0,0,1], [0,2,0], [-1,0,0]])
>>> B = A.log()
>>> B.exp()
Traceback (most recent call last):
  ...
NotImplementedError: unable to compute matrix exponential
```

But it's not perfect...

```
>>> A = ca_mat([[0,0,1], [0,2,0], [-1,0,0]])
>>> B = A.log()
>>> B.exp()
Traceback (most recent call last):
  ...
NotImplementedError: unable to compute matrix exponential
```

Equality is always decidable over $\overline{\mathbb{Q}}$

Better algorithms for transcendentals: hard



https://ask.sagemath.org/question/52653/equality-of-algeb

Equality of algebraic numbers given by huge symbolic expressions

2

large | symbolic | expressions | algebraic_number

I calculated a matrix whose first entry is a huge numerical value:

asked Jul 24 '0
creyesm1992
61 •5

updated Aug 5 '0
slelievre
12431 •11 •121 •247
http://carva.org/samue...

```
N = 1/16*(44*(7*sqrt(2) - 10)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26) + 2*(11*(7*sqrt(2) - 1
0)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26) - 10*(63*sqrt(2) - 89)*sqrt(sqrt(2) + 2) - (3*(3*s
qrt(2) - 4)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26) - (85*sqrt(2) - 122)*sqrt(sqrt(2) + 2))*s
qrt(-sqrt(2) + 2) + 2*((3*(3*sqrt(2) - 4)*sqrt(-17*sqrt(2) + 26) - 85*sqrt(2) + 122)*sqrt(-sqr
t(2) + 2) - 11*(7*sqrt(2) - 10)*sqrt(-17*sqrt(2) + 26) + 630*sqrt(2) - 890)*sqrt(sqrt(sqrt(2)
+ 2) - 1))*sqrt(3*sqrt(2) + sqrt(-17*sqrt(2) + 26) - 3) - 40*(63*sqrt(2) - 89)*sqrt(sqrt(2) +
2) - 4*(3*(3*sqrt(2) - 4)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26) - (85*sqrt(2) - 122)*sqrt(s
qrt(2) + 2))*sqrt(-sqrt(2) + 2) + (22*(5*sqrt(2) - 7)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26)
+ (11*(5*sqrt(2) - 7)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26) - 5*(89*sqrt(2) - 126)*sqrt(sqr
t(2) + 2) - (3*(2*sqrt(2) - 3)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26) - (61*sqrt(2) - 85)*sq
rt(sqrt(2) + 2))*sqrt(-sqrt(2) + 2) + 2*((3*(2*sqrt(2) - 3)*sqrt(-17*sqrt(2) + 26) - 61*sqrt
(2) + 85)*sqrt(-sqrt(2) + 2) - 11*(5*sqrt(2) - 7)*sqrt(-17*sqrt(2) + 26) + 445*sqrt(2) - 630)*
sqrt(sqrt(sqrt(2) + 2) - 1))*sqrt(3*sqrt(2) + sqrt(-17*sqrt(2) + 26) - 3) - 10*(89*sqrt(2) - 1
26)*sqrt(sqrt(2) + 2) - 2*(3*(2*sqrt(2) - 3)*sqrt(sqrt(2) + 2)*sqrt(-17*sqrt(2) + 26) - (61*sq
rt(2) - 85)*sqrt(sqrt(2) + 2))*sqrt(-sqrt(2) + 2) + 4*((3*(2*sqrt(2) - 3)*sqrt(-17*sqrt(2) + 2
6) - 61*sqrt(2) + 85)*sqrt(-sqrt(2) + 2) - 11*(5*sqrt(2) - 7)*sqrt(-17*sqrt(2) + 26) + 445*sqr
t(2) - 630)*sqrt(sqrt(sqrt(2) + 2) - 1))*sqrt(-12*sqrt(2) - 2*sqrt(-sqrt(2) + 2) - 2*sqrt(-17*
```

(This goes on for 12 screens.)

https://ask.sagemath.org/question/52653/equality-of-algeb

I have to check if this value is equal to `-(1-(abs(M))^2)^2`.

where

```
M = -(4*(6*sqrt(2) + sqrt(-sqrt(2) + 2) + sqrt(-17*sqrt(2) + 26) - 8)*sqrt(3*sqrt(2) + sqrt(-s
qrt(2) + 2) - 5) - sqrt(3*sqrt(2) + sqrt(-17*sqrt(2) + 26) - 3)*(-24*I*sqrt(2) - 4*I*sqrt(-sqr
t(2) + 2) - 4*I*sqrt(-17*sqrt(2) + 26) + 32*I) - ((sqrt(2)*sqrt(-sqrt(2) + 2) + sqrt(2)*sqrt(-
17*sqrt(2) + 26) - 8*sqrt(2) + 12)*sqrt(3*sqrt(2) + sqrt(-sqrt(2) + 2) - 5) + (I*sqrt(2)*sqrt
(-sqrt(2) + 2) + I*sqrt(2)*sqrt(-17*sqrt(2) + 26) - 8*I*sqrt(2) + 12*I)*sqrt(3*sqrt(2) + sqrt
(-17*sqrt(2) + 26) - 3))*sqrt(-12*sqrt(2) - 2*sqrt(-sqrt(2) + 2) - 2*sqrt(-17*sqrt(2) + 26) +
24) - ((24*I*sqrt(2) + 4*I*sqrt(-17*sqrt(2) + 26) - 32*I)*sqrt(-sqrt(2) + 2) + 8*I*(3*sqrt(2)
- 4)*sqrt(-17*sqrt(2) + 26) - 228*I*sqrt(2) + 328*I)*sqrt(sqrt(sqrt(2) + 2) - 1))/(4*(6*sqrt
(2) + sqrt(-sqrt(2) + 2) + sqrt(-17*sqrt(2) + 26) - 8)*sqrt(3*sqrt(2) + sqrt(-17*sqrt(2) + 26)
- 3)*sqrt(sqrt(sqrt(2) + 2) - 1) + sqrt(3*sqrt(2) + sqrt(-sqrt(2) + 2) - 5)*(-24*I*sqrt(2) - 4
*I*sqrt(-sqrt(2) + 2) - 4*I*sqrt(-17*sqrt(2) + 26) + 32*I)*sqrt(sqrt(sqrt(2) + 2) - 1) - 4*(6*
sqrt(2) + sqrt(-17*sqrt(2) + 26) - 8)*sqrt(-sqrt(2) + 2) + ((I*sqrt(2)*sqrt(-sqrt(2) + 2) + I*
sqrt(2)*sqrt(-17*sqrt(2) + 26) - 8*I*sqrt(2) + 12*I)*sqrt(3*sqrt(2) + sqrt(-sqrt(2) + 2) - 5)*
sqrt(sqrt(sqrt(2) + 2) - 1) - (sqrt(2)*sqrt(-sqrt(2) + 2) + sqrt(2)*sqrt(-17*sqrt(2) + 26) - 8
*sqrt(2) + 12)*sqrt(3*sqrt(2) + sqrt(-17*sqrt(2) + 26) - 3)*sqrt(sqrt(sqrt(2) + 2) - 1))*sqrt
(-12*sqrt(2) - 2*sqrt(-sqrt(2) + 2) - 2*sqrt(-17*sqrt(2) + 26) + 24) - 8*(3*sqrt(2) - 4)*sqrt
(-17*sqrt(2) + 26) + 228*sqrt(2) - 328)
```

so i run the following cell:

```
bool(N == -(1 - abs(M)^2)^2)
```

Sadly it keeps loading for hours (at 6 hours I stopped the kernel), and i do not know if this last cell gives me true of false.

```
fredrik@agm:~/src/calcium$ build/examples/huge_expr -ca
Evaluating N...
cpu/wall(s): 0.204 0.203
Evaluating M...
cpu/wall(s): 0.03 0.03
Evaluating E = -(1-|M|^2)^2...
cpu/wall(s): 0.01 0.01
N ~ -0.16190853053311203695842869991458578203473645660641
E ~ -0.16190853053311203695842869991458578203473645660641
Testing E = N...
cpu/wall(s): 89.161 89.173

Equal = T_TRUE

Total: cpu/wall(s): 89.405 89.418
virt/peak/res/peak(MB): 60.31 68.37 34.34 42.30
```

Some examples

$$\mathbf{x} - \mathrm{DFT}^{-1}(\mathrm{DFT}(\mathbf{x})) = \mathbf{0}$$

Test vector: $x_n = \sqrt{n+2},\ \ n = 0, \ldots, N-1$

N	Sage $\overline{\mathbb{Q}}$	Sage `SR`	SymPy	Maple	Mathematica	Calcium
8	5.3	0.50	2.8	0.046	0.11	0.017
16	$> 10^3$	46	24	0.26	0.58	0.090
20	$> 10^3$	154	fail	1.1	2.3	0.17
100	$> 10^3$	fail	fail	$> 10^3$	> 60	38

More test vectors in paper:

$$n+2 \quad \log(n+2) \quad e^{2\pi i/(n+2)} \quad \frac{1}{1+(n+2)\pi} \quad \frac{1}{1+\sqrt{n+2}\pi}$$

Finitely generated subfields of $\mathbb{C}$

Definitions

- $K = \mathbb{Q}(a_1, \ldots, a_n), \quad a_1, \ldots, a_n \in \mathbb{C}$
  Field generated by extension numbers a_k

Finitely generated subfields of $\mathbb{C}$

Definitions

- $K = \mathbb{Q}(a_1, \ldots, a_n), \quad a_1, \ldots, a_n \in \mathbb{C}$
  Field generated by extension numbers a_k
- $R = \mathbb{Q}[X_1, \ldots, X_n]$
  Polynomial ring

Finitely generated subfields of $\mathbb{C}$

Definitions

- $K = \mathbb{Q}(a_1, \ldots, a_n), \quad a_1, \ldots, a_n \in \mathbb{C}$
  Field generated by extension numbers a_k
- $R = \mathbb{Q}[X_1, \ldots, X_n]$
  Polynomial ring
- $\mu : R \to \mathbb{C}, \quad X_k \mapsto a_k$
  Numerical embedding (evaluation homomorphism)

Finitely generated subfields of $\mathbb{C}$

Definitions

- $K = \mathbb{Q}(a_1, \ldots, a_n), \quad a_1, \ldots, a_n \in \mathbb{C}$
  Field generated by extension numbers a_k
- $R = \mathbb{Q}[X_1, \ldots, X_n]$
  Polynomial ring
- $\mu : R \to \mathbb{C}, \quad X_k \mapsto a_k$
  Numerical embedding (evaluation homomorphism)
- $I = \ker \mu = \{f \in R : f(a_1, \ldots, a_n) = 0\}$
  The ideal of all algebraic relations among $a_1, \ldots, a_n$ over $\mathbb{Q}$

Finitely generated subfields of $\mathbb{C}$

Definitions

- $K = \mathbb{Q}(a_1, \ldots, a_n), \quad a_1, \ldots, a_n \in \mathbb{C}$
  Field generated by extension numbers a_k
- $R = \mathbb{Q}[X_1, \ldots, X_n]$
  Polynomial ring
- $\mu : R \to \mathbb{C}, \quad X_k \mapsto a_k$
  Numerical embedding (evaluation homomorphism)
- $I = \ker \mu = \{f \in R : f(a_1, \ldots, a_n) = 0\}$
  The ideal of all algebraic relations among $a_1, \ldots, a_n$ over $\mathbb{Q}$
- $K_{\text{formal}} = \text{Frac}(R/I)$
  Formal field

Finitely generated subfields of $\mathbb{C}$

Definitions

- $K = \mathbb{Q}(a_1, \ldots, a_n), \quad a_1, \ldots, a_n \in \mathbb{C}$
  Field generated by extension numbers a_k
- $R = \mathbb{Q}[X_1, \ldots, X_n]$
  Polynomial ring
- $\mu : R \to \mathbb{C}, \quad X_k \mapsto a_k$
  Numerical embedding (evaluation homomorphism)
- $I = \ker \mu = \{f \in R : f(a_1, \ldots, a_n) = 0\}$
  The ideal of all algebraic relations among $a_1, \ldots, a_n$ over $\mathbb{Q}$
- $K_{\text{formal}} = \text{Frac}(R/I)$
  Formal field

Theorem: $K \cong K_{\text{formal}}$

Finitely generated subfields of $\mathbb{C}$

Definitions

- $K = \mathbb{Q}(a_1, \ldots, a_n), \quad a_1, \ldots, a_n \in \mathbb{C}$
  Field generated by extension numbers a_k
- $R = \mathbb{Q}[X_1, \ldots, X_n]$
  Polynomial ring
- $\mu : R \to \mathbb{C}, \quad X_k \mapsto a_k$
  Numerical embedding (evaluation homomorphism)
- $I = \ker \mu = \{f \in R : f(a_1, \ldots, a_n) = 0\}$
  The ideal of all algebraic relations among $a_1, \ldots, a_n$ over $\mathbb{Q}$
- $K_{\text{formal}} = \text{Frac}(R/I)$
  Formal field

Theorem: $K \cong K_{\text{formal}}$

Theorem: if $I = \langle f_1, \ldots, f_r \rangle$ is known, K is an effective field (proof: Gröbner bases)

Notable special cases

The trivial field $K = \mathbb{Q}$

Notable special cases

The trivial field $K = \mathbb{Q}$

Transcendental number fields

$$K = \mathbb{Q}(a_1, \ldots, a_n) \cong \mathbb{Q}(X_1, \ldots, X_n),$$
$$a_1, \ldots, a_n \text{ algebraically independent over } \mathbb{Q}$$

Notable special cases

The trivial field $K = \mathbb{Q}$

Transcendental number fields

$$K = \mathbb{Q}(a_1, \ldots, a_n) \cong \mathbb{Q}(X_1, \ldots, X_n),$$
$$a_1, \ldots, a_n \text{ algebraically independent over } \mathbb{Q}$$

Algebraic number fields

$$K = \mathbb{Q}(a) \;\cong\; \mathbb{Q}[X]/\langle f(X) \rangle$$

Notable special cases

The trivial field $K = \mathbb{Q}$

Transcendental number fields

$$K = \mathbb{Q}(a_1, \ldots, a_n) \cong \mathbb{Q}(X_1, \ldots, X_n),$$
$$a_1, \ldots, a_n \text{ algebraically independent over } \mathbb{Q}$$

Algebraic number fields

$$K = \mathbb{Q}(a) \;\cong\; \mathbb{Q}[X]/\langle f(X) \rangle$$

Mixed fields

Example: $K = \mathbb{Q}(\log(i), \pi, i) \cong \mathrm{Frac}(\mathbb{Q}[X_1, X_2, X_3]/I)$
where $I = \langle 2X_1 - X_2 X_3,\; X_3^2 + 1 \rangle$

Defining extension numbers

- Algebraic numbers (e.g. i, $\sqrt{2}$, $e^{2\pi i/5}$): canonical representation by minimal polynomial over $\mathbb{Q}$ + root enclosure
- Symbolic functions and constants: $f(z_1, \ldots, z_n)$
- Black-box numerical evaluation

Not so fast...

Problem: we may not be able to compute the I in

$$\mathbb{Q}(a_1, \ldots, a_n) \cong \mathrm{Frac}(\mathbb{Q}[X_1, \ldots, X_n]/I)$$

Example:

- $\mathbb{Q}(\pi) \cong \mathbb{Q}(X_1)$
- $\mathbb{Q}(e) \cong \mathbb{Q}(X_2)$
- Is $\mathbb{Q}(\pi, e) \cong \mathbb{Q}(X_1, X_2)$?
  (Open problem: Schanuel's conjecture.)

Example:

- $\mathbb{Q}(a_1, \ldots, a_n)$ with algebraic $a_k \to$ impossibly large polynomials

Working with an incomplete ideal

Instead of computing I, compute some *reduction ideal* $I_{\text{red}} \subseteq I$:

$$\mathbb{Q}(a_1, \ldots, a_n) \overset{?}{\cong} \text{Frac}(\mathbb{Q}[X_1, \ldots, X_n]/I_{\text{red})}$$

Can use the map μ (numerical evaluation) as certificate of nonvanishing for given $z \in K$.

Working with an incomplete ideal

Instead of computing I, compute some *reduction ideal* $I_{\text{red}} \subseteq I$:

$$\mathbb{Q}(a_1, \ldots, a_n) \overset{?}{\cong} \text{Frac}(\mathbb{Q}[X_1, \ldots, X_n]/I_{\text{red}})$$

Can use the map μ (numerical evaluation) as certificate of nonvanishing for given $z \in K$.

Algorithm: test if $z = 0$ where $z \cong p/q$

- If $p \equiv 0 \bmod I_{\text{red}}$, return True.
- If it can be certified that $I_{\text{red}} = I$, return False.
- Using ball arithmetic, compute an enclosure E with $\mu(p) \in E$. If $0 \notin E$, return False.
- Attempt to find and prove a new set of relations J with $J \subseteq I$, and set $I_{\text{red}} \leftarrow I_{\text{red}} \cup J$. Repeat.

Ideal construction

Heuristics to construct I_{red}:

- Direct algebraic relations: $a_k \in \overline{\mathbb{Q}}$, $a_k = \sqrt{z}$, etc.
- Log-linear relations

$$m_1 \log(a_1) + \ldots + m_k \log(a_k) = 0$$

  - LLL gives basis matrix of potential relations
  - Verification through recursive computations in simpler fields
  - Same idea for multiplicative relations $a_1^{m_1} \cdots a_k^{m_k} = 1$
- Functional equations: $\Gamma(z+1) = z\Gamma(z)$, etc.
- Other algebraic relations: resultants, Vieta's formulas, etc.

Quotient ring and fraction field arithmetic

Practical concerns about implementing arithmetic in $\mathbb{Q}(a_1, \ldots, a_n) \cong \mathrm{Frac}(\mathbb{Q}[X_1, \ldots, X_n]/I)$

- Ordering monomials: lex, deglex, etc.
  - Cost of Gröbner basis computation, size of polynomials

- Ordering extension numbers: $e^\pi \succ \pi \succ i$

- Normalizing fractions
  - Always remove content in $\mathbb{Q}[X_1, \ldots, X_n]$?
  - Rationalizing denominators

Non-canonical fractions

Problem: f, g reduced modulo I and coprime in $\mathbb{Q}[X_1, \ldots, X_n]$
$\not\Longrightarrow \frac{f}{g}$ in canonical form

```
>>> a = exp(pi)
>>> b = exp(-pi)
>>> a*b
1
```

```
>>> a
23.1407 {a where ...}
>>> (a**3 - 2*a + b) / (a**2 + b**2 - 2)
23.1407 {(a^3-2*a+b)/(a^2+b^2-2) where ...}
```

```
>>> (a**3 - 2*a + b) / (a**2 + b**2 - 2) - a
0
```

Solutions and workarounds

- Always rationalize the denominator
  - Practical in simple cases
- Compute polynomial GCD over $\mathbb{Q}(\alpha)$ instead of $\mathbb{Q}$
  - Only applicable in some cases, potentially expensive
- General algorithm for simplifying or canonicalizing fractions modulo an ideal: Monagan and Pearce (2006)
  - Uses Gröbner bases over modules, potentially expensive
- Use algorithms that minimize divisions

Determinant of $A_{i,j} = \sqrt{i+j-1}, 1 \le i,j \le 5$

$$\mathbb{Q}(\sqrt{7}, \sqrt{6}, \sqrt{5}, \sqrt{3}, \sqrt{2}) \overset{?}{\cong} \mathrm{Frac}(\mathbb{Q}[a,b,c,d,e]/\langle a^2-7, b^2-6, c^2-5, d^2-3, e^2-2, b-de\rangle)$$

Gaussian elimination:

```
(156829688*a*c*d*e-221693656*a*c*d+271638392*a*c*e-383986048*a*c
    +274164856*a*d*e-387945384*a*d+474865368*a*e-671936784*a+361353464*
    c*d*e-510531104*c*d+625886152*c*e-884270248*c+959654264*d*e
    -1358274640*d+1662163432*e-2352590040) / (18200*a*c*d*e-25732*a*c*d
    +31512*a*c*e-44565*a*c+324056*a*d*e-458284*a*d+561288*a*e-793807*a
    +847420*c*d*e-1198107*c*d+1467772*c*e-2075132*c+1068396*d*e
    -1511729*d+1850596*e-2618400)
```

Bareiss algorithm (fraction-free Gauss):

```
(-28*a*c*d*e+48*a*c*d+20*a*c*e-116*a*c+460*a*d*e-520*a*d+332*a*e-532*a
    +348*c*d*e-516*c*d-332*c*e+120*c+548*d*e-388*d+1660*e-2144) / (c*d
    -2*c+4*d*e-3*d-4)
```

Cofactor expansion or Berkowitz algorithm:

```
-4*a*c*d-20*a*c*e-24*a*c-4*a*d*e+8*a*d+136*a-28*c*d*e-116*c*d-88*c*e+64*
    c+112*d*e+164*d-60*e+244
```

Things to do

- Lots of basic implementation work
- Efficient Gröbner basis computation
- Better algorithms for dealing with fractions fields
- Better algorithms for algebraic number fields
- Implement Richardson's algorithm (perhaps simplified)
- Good algorithms for real/complex parts, real trigonometric functions, etc.
- Speed up integer relations
- Efficient extension $\mathbb{Q}(a_1, \ldots, a_{n-1}) \to \mathbb{Q}(a_1, \ldots, a_n)$