Newton sums
for an effective formalization of algebraic numbers

Cyril Cohen, Boris Djalal
Inria Sophia Antipolis – Méditerranée, France

project funded by Microsoft Research Inria Joint Center

January 12, 2016

Motivation

Applications:

- semialgebraic sets
- computer algebra
- formalization of robotics.

Motivation

Applications:

- semialgebraic sets
- computer algebra
- formalization of robotics.

Concerns: efficiency + certification.

Our goals:

- formalize efficient algorithms
  to compute real algebraic numbers in Coq
- provide computable versions of these algorithms.

Benefits of algebraic numbers:

- field structure
- decidable equality
- countable.

Introduction: what is an algebraic number ?

- An algebraic number is a number which is the root of a polynomial with rational coefficients
- for example, $\sqrt{2}$ is an algebraic number because it is a root of the polynomial $X^2 - 2$
- π is not an algebraic number.
- We denote algebraic numbers by $\overline{\mathbb{Q}}$.

Representation of algebraic numbers

- We represent an algebraic number by:
  - a polynomial
  - a piece of information to retain one root of the polynomial.
- For example, we can represent $\sqrt{2}$ by:
  - $X^3 - X^2 - 2X + 2$
  - the interval $[1.3, 2]$
  - a proof that P has exactly one root in $[1.3, 2]$.

- All operations
  (addition, multiplication, inversion and comparison)
  must be based on our representation.

Representation of algebraic numbers

- We represent an algebraic number by:
  - a polynomial
  - a piece of information to retain one root of the polynomial.
- For example, we can represent $\sqrt{2}$ by:
  - $X^3 - X^2 - 2X + 2$
  - the interval $[1.3, 2]$
  - a proof that P has exactly one root in $[1.3, 2]$.
- All operations
  (addition, multiplication, inversion and comparison)
  must be based on our representation.
- Let $a, b \in \overline{\mathbb{Q}}$ and $P, Q \in \mathbb{Q}[X]$ such that $P(a) = 0$, $Q(b) = 0$.
  We want to compute polynomials R_1 and R_2 such that
  $R_1(a + b) = 0$ and $R_2(a \times b) = 0$.

Composed sum and composed product

- $\alpha, \beta, a, b \in \overline{\mathbb{Q}}$
- a is a root of $P \in \mathbb{Q}[X]$: $P(a) = 0$
- b is a root of $Q \in \mathbb{Q}[X]$: $Q(b) = 0$.
- $\text{roots}(P)$ denotes the multiset of roots of P in $\overline{\mathbb{Q}}$

Composed sum and composed product

- $\alpha, \beta, a, b \in \overline{\mathbb{Q}}$
- a is a root of $P \in \mathbb{Q}[X]$: $P(a) = 0$
- b is a root of $Q \in \mathbb{Q}[X]$: $Q(b) = 0$.
- roots(P) denotes the multiset of roots of P in $\overline{\mathbb{Q}}$
- The number $a + b$ is a root of $\displaystyle\prod_{\substack{\alpha \in \text{roots}(P) \\ \beta \in \text{roots}(Q)}} (X - (\alpha + \beta))$
- we note this polynomial: $P \oplus Q$
- we call it the "composed sum" of P and Q.

- Coefficients of $P \oplus Q$ are a symmetric function of its roots
- thus, according to the theorem of symmetric polynomials, the coefficients of $P \oplus Q$ belong to $\mathbb{Q}$.

Composed sum and composed product

- $\alpha, \beta, a, b \in \overline{\mathbb{Q}}$
- a is a root of $P \in \mathbb{Q}[X]$: $P(a) = 0$
- b is a root of $Q \in \mathbb{Q}[X]$: $Q(b) = 0$.
- roots(P) denotes the multiset of roots of P in $\overline{\mathbb{Q}}$
- The number $a + b$ is a root of $\prod_{\substack{\alpha \in \text{roots}(P) \\ \beta \in \text{roots}(Q)}} (X - (\alpha + \beta))$
- we note this polynomial: $P \oplus Q$
- we call it the "composed sum" of P and Q.

- Coefficients of $P \oplus Q$ are a symmetric function of its roots
- thus, according to the theorem of symmetric polynomials, the coefficients of $P \oplus Q$ belong to $\mathbb{Q}$.

- Similarly, we define the composed product of P and Q.

Newton representation

- Our work is based on *Algorithmique efficace pour des opérations de base en Calcul formel* - Alin Bostan (2003).
- Definition: $\mathcal{N} : \mathbb{Q}[X] \to \mathbb{Q}[\![X]\!]$

$$P \mapsto \mathcal{N}(P) = \sum_{i=0}^{\infty} \left(\sum_{\alpha \in \mathrm{roots}(P)} \alpha^i \right) X^i$$

- we call it the Newton representation of P.
- In pratice, we only need the first terms of $\mathcal{N}(P)$
- the truncated power series can be computed without knowing α's.

Newton transformations

[Alin Bostan 2003] provides a method to:

- transform a polynomial into a power series with $\mathcal{N}$.
- transform back from $\mathcal{N}(P)$ into P.

[Alin Bostan 2003] defines:

- an addition $\boxplus$ in the Newton space
- a multiplication $\boxtimes$ in the Newton space.

We formally described the algorithms and proved these statements:

- $\mathcal{N}^{-1}(\mathcal{N}(P)) = P$ when $P(0) \neq 0$
- $P \oplus Q = \mathcal{N}^{-1}(\mathcal{N}(P) \boxplus \mathcal{N}(Q))$
- $P \otimes Q = \mathcal{N}^{-1}(\mathcal{N}(P) \boxtimes \mathcal{N}(Q))$.

Newton transformations

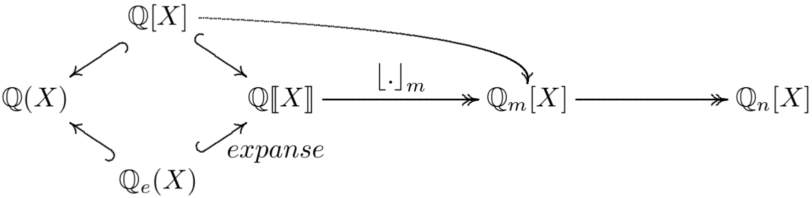
$$\mathcal{N}(P) = \frac{\mathrm{rev}(P')}{\mathrm{rev}(P)}$$

$$\mathcal{N}^{-1}(f) = \mathrm{rev}\left(\exp\left(\int \frac{1}{X}(f_0 - f)\right)\right)$$

Need for:

- rev: reverse the coefficients of a polynomial
- exponential of FPS
- primitive $\int$ on FPS

High-level picture of involved objects



- $\mathbb{Q}[[X]]$ denotes the ring of formal power series
- $\mathbb{Q}_m[X]$ denotes the ring of truncated formal power series
- $\mathbb{Q}_e(X)$ denotes the ring of expansible rational fractions
  examples: $\frac{1}{1-X}$ expanses to $1 + X + X^2 + \ldots$ but $\frac{1}{X} \notin \mathbb{Q}_e(X)$
- $\mathbb{Q}[X] \to \mathbb{Q}_m[X]$ denotes the canonical surjection
  which sends any polynomial P to P modulo X^{m+1}.

Contributions

We needed to develop the following notions:

- truncated formal power series
    - derivative
    - primitive
    - composition
    - logarithm
    - exponential
- fractions of polynomials
- expansible rational fractions.

Truncated formal power series (TFPS)

- We formalize TFPS_m with a Record in Coq:

```
Record tfps := MkTfps
{
  truncated_tfps :> {poly K};
  _ : size truncated_tfps <= m.+1
}.
```

- polynomial + proof on the degree
- dependent type allow us to create such a pair
- our Record is a subtype of polynomials because we can decide whether the size is less than $m + 1$.

Results on TFPS

- Build a TFPS_m from the proof that $\mathrm{size}\,(P \mod X^{m+1}) \leq m+1$.

- Build a TFPS from its coefficients.

Results on TFPS

- Build a TFPS_m from the proof that size $(P \mod X^{m+1}) \leq m+1$.
- Build a TFPS from its coefficients.
- structure on TFPS_m
  - commutative ring
  - in TFPS_3, $X^2 \cdot X^2 = 0 \pmod{X^4}$
- derivative: $\mathbb{Q}_{m+1}[X] \longrightarrow \mathbb{Q}_m[X]$
- primitive: $\mathbb{Q}_m[X] \longrightarrow \mathbb{Q}_{m+1}[X]$
- logarithm, exponential: from a subtype of $\mathbb{Q}_m[X]$ to $\mathbb{Q}_m[X]$.

TFPS: exponential and logarithm

Let f be a TFPS_m.

- If $f_0 = 0$ we define:

$$\exp(f) = \sum_{i=0}^{m} \frac{f^i}{i!}$$

- If $f_0 = 1$ we define:

$$\log(f) = -\sum_{i=1}^{m} \frac{(1-f)^i}{i}.$$

TFPS: derivative

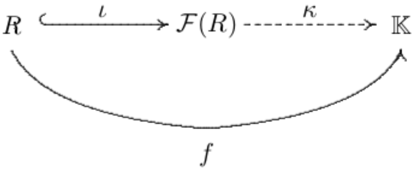
$\forall m \in \mathbb{N}, \ \ \forall f, g \in K_{m+1}[X]$

- $(f+g)' =_{\boldsymbol{K_m[X]}} f' + g'$
- $(f \cdot g)' =_{\boldsymbol{K_m[X]}} f' \cdot \lfloor g \rfloor_m + \lfloor f \rfloor_m \cdot g'$
- if $f_0 = 0$: $\quad (\exp f)' =_{\boldsymbol{K_m[X]}} f' \cdot \lfloor \exp(f) \rfloor_m$
- if $f_0 = 1$: $\quad (\log f)' =_{\boldsymbol{K_m[X]}} \dfrac{f'}{\lfloor f \rfloor_m}.$

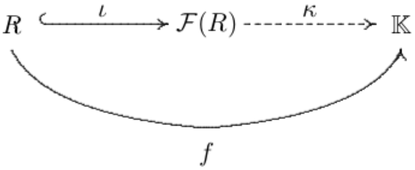
Universal property of the field of fractions

R is an integral domain.
There is a field $\mathcal{F}(R)$ and a ring morphism ι satisfying:

for any field $\mathbb{K}$ and injective ring morphism f from R to $\mathbb{K}$,
there is a unique ring morphism κ s.t. our diagram commutes.



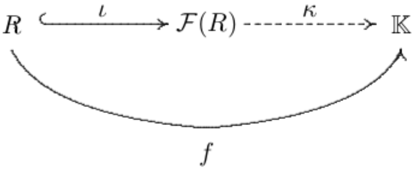
Universal property of field of fractions: how κ is defined ?



Let $\frac{u}{v} \in \mathcal{F}(R)$:

- by definition of $\mathcal{F}(R)$, $v \neq 0$
- since $v \neq 0$ and f is an injective ring morphism, $f(v) \neq 0$
- thus we can compute the inverse of $f(v)$ in $\mathbb{K}$
- we set $\kappa(\frac{u}{v}) = \frac{f(u)}{f(v)}$.

Universal property of field of fractions: how κ is defined ?



Let $\frac{u}{v} \in \mathcal{F}(R)$:

- by definition of $\mathcal{F}(R)$, $v \neq 0$
- since $v \neq 0$ and f is an injective ring morphism, $f(v) \neq 0$
- thus we can compute the inverse of $f(v)$ in $\mathbb{K}$
- we set $\kappa(\frac{u}{v}) = \frac{f(u)}{f(v)}$.

We generalize the condition on f:

- we just require $f(v) \neq 0$, not f injectivity.

Regular morphism

The computability of κ is guaranted when these three points are satisfied:

- f is computable
- given $x \in \mathcal{F}(R)$ it is decidable
  - whether there is a regular representation for x
  - whether x is regular for f
  - whether $\exists u, v \in R,\ f(v) \neq 0$ and $x = \frac{u}{v}$
- when x is regular for f, a regular representation of x is computable.

Regular morphism

The computability of κ is guaranted when
these three points are satisfied:

- f is computable
- given $x \in \mathcal{F}(R)$ it is decidable
  - whether there is a regular representation for x
  - whether x is regular for f
  - whether $\exists u, v \in R,\ f(v) \neq 0$ and $x = \frac{u}{v}$
- when x is regular for f,
  a regular representation of x is computable.

We say that f is regular.
If f is injective then f is regular and all $x \in \mathcal{F}(R)$ are regular for f.

Abstract evaluation results

We derive formally the following results:

- $\kappa(0) = 0$
- $\kappa(1) = 1$
- $\forall x \in \mathcal{F}(R), \ \ \kappa(-x) = -\kappa(x)$
- $\forall x, y$ regular for f, $\ \ \kappa(x + y) = \kappa(x) + \kappa(y)$
- $\forall x, y$ regular for f, $\ \ \kappa(x \cdot y) = \kappa(x) \cdot \kappa(y)$
- $\forall x, y \in \mathcal{F}(R), \ \ \kappa(y) \neq 0 \implies \kappa(\frac{x}{y}) = \frac{\kappa(x)}{\kappa(y)}$
- if f is injective then κ is a ring morphism

This interface is then instanciated twice in our code.

First case: evaluating fractions of polynomials

- The evaluation of $X^2 - 2$ in 3 gives 7
- the evaluation of $\frac{X^2 - 2}{X + 5}$ in 3 gives $\frac{7}{8}$
- the evaluation of $\frac{X^2 - 2}{X - 3}$ in 3 is not defined because we cannot find a regular representation (3 is a pole)
- the evaluation of $\frac{X^2 - 3X}{X^2 - X - 6}$ in 3 is defined:
  - we move to the equivalent regular representation $\frac{X}{X + 2}$
  - it gives $\frac{3}{5}$.

Abstraction over the evaluation on fractions of polynomials

- $\mathbb{K}$ is a field
- $\mathbb{K}[X]$ is an integral domain R
- $\mathbb{K}(X)$ is the field of fractions of R, noted $\mathcal{F}(R)$
- $f\colon\ R \longrightarrow \mathbb{K}$ is the evaluation of polynomials in $a = 3$.

- Our evaluation of fractions of polynomials is the map:
  $\kappa : \mathcal{F}(R) \longrightarrow \mathbb{K}$
  $$\kappa(x) = \begin{cases} \frac{f(u)}{f(v)} & \text{if } x \text{ can be written as } \frac{u}{v} \text{ with } f(v) \neq 0 \\ \textit{undefined} & \text{otherwise.} \end{cases}$$

- Note that f is parameterized by an element $a \in \mathbb{K}$.

Second case: lifting from $F(X)$ to $L(X)$

- $F \hookrightarrow L$ is a field extension.
- we know how to lift from $F[X]$ to $L(X)$
- problem: we want to lift any element of $x \in F(X)$ to $L(X)$.

Solution:

- x writes as $\frac{u}{v}$ with $u \in F[X]$, $v \in F[X]$
- we lift u and v and perform a division.

Abstraction over the lifting from $F(X)$ to $L(X)$

- $F[X]$ is an integral domain R
- $F(X)$ is the field of fractions of R, noted $\mathcal{F}(R)$
- $L(X)$ is a field $\mathbb{K}$
- $f\colon\ R \longrightarrow \mathbb{K}$ is the lifting from $F[X]$ to $L(X)$.

- Our lifting function from $F(X)$ to $L(X)$ is the map:
  $\kappa : \mathcal{F}(R) \longrightarrow \mathbb{K}$
  $$\kappa(x) = \begin{cases} \frac{f(u)}{f(v)} & \text{if } x \text{ can be written as } \frac{u}{v} \text{ with } f(v) \neq 0 \\ \textit{undefined} & \text{otherwise.} \end{cases}$$

- Note that here f is injective.
- Thus, κ is defined on whole $\mathcal{F}(R)$.

Sum-up of our contributions

- Formalization of truncated power series
  - $+$, x, commutative ring
- Newton space:
  - Newton transformation in both directions
  - $\boxplus$ and $\boxtimes$ in Newton space
  - morphism lemmas
- formal proofs of correctness
- abstract evaluation of fractions.

Related work

During our formalization, we had to use existing concepts from Mathematical Components:

- polynomials
- polynomial divisibility
- finite iterations of operations (bigop.v)
- binomial numbers
- fractions.

We also used developments for elliptic curves from Pierre-Yves Strub (xseq, polyorder, polyall, polydec):

- polynomials and multiplicity
- roots of polynomials and equality up to a permutation.

Future work

- select one root of a polynomial
  - Thom encoding

    Algorithms in Real Algebraic Geometry
    - Saugata Basu, Richard Pollack, Marie-Françoise Roy (2011)
  - Newton method
    - work of Iona Pasca on multivariate analysis
- run computable versions of the algorithms inside Coq.
  - CoqEAL `https://github.com/CoqEAL/CoqEAL`

Questions