Toward a formal proof that zeta(3) is irrational

joint work with Frédéric Chyzak (Inria) and Thomas Sibut-Pinote (ENS Lyon)

Assia Mahboubi

INRIA Microsoft Research Joint Centre (France)
INRIA Saclay – Île-de-France
Lix, École Polytechnique, Palaiseau

July 26th 2012

“A proof that Euler missed... (A. Van de Poorten, 1979)”

Theorem (Apéry, 1978) : $\zeta(3) := \sum_{k=0}^{+\infty} \frac{1}{n^3}$ is irrational.

Sketch of the proof

In order to prove that $\zeta(3) \notin \mathbb{Q}$ we show that otherwise we could exhibit a sequence S_n such that:

- $\forall n, \quad S_n$ is an integer
- $\forall n, \quad S_n > 0$
- $\lim\limits_{n \to \infty} S_n = 0$

Sketch of the proof

We construct two (complicated) sequences (a_n) and (b_n) such that:

$$a_n\zeta(3) - b_n \to 0 \quad \text{and} \quad \forall n, a_n \in \mathbb{Z}^* \quad b_n \in \mathbb{Q}^*$$

Now in fact if we pose $d_n := lcm(1, \ldots, n)$ we even have:

$$d_n^3(a_n\zeta(3) - b_n) \to 0$$

Sketch of the proof

We construct two (complicated) sequences (a_n) and (b_n) such that:

$$a_n\zeta(3) - b_n \to 0 \quad \text{and} \quad \forall n, a_n \in \mathbb{Z}^* \quad b_n \in \mathbb{Q}^*$$

Now in fact if we pose $d_n := lcm(1, \ldots, n)$ we even have:

$$d_n^3(a_n\zeta(3) - b_n) \to 0$$

In fact:

$S_n := d_n^3(a_n\zeta(3) - b_n)$ is the desired absurd sequence.

Steps in the proof

- $a_n\zeta(3) - b_n \to 0$

  elementary.

- $d_n^3 b_n \in \mathbb{Z}$:

  considered as elementary arithmetic.

- $d_n \sim e^n$:

  considered as standard.

- $a_n\zeta(3) - b_n > 0$:

  study of the asymptotic of a remainder since $a_n\zeta(3) - b_n \to 0$

- $d_n^3(a_n\zeta(3) - b_n) \to 0$:

  study of the asymptotic of the sequence $(a_n\zeta(3) - b_n)$

The two last steps are based on the proof that (a_n) and (b_n) satisfy a common (linear with polynomial coefficients) recurrence relation.

Now the real sequences

We pose:

- $z_n := \sum_{k=1}^{n} \frac{1}{k^3}$
- $c_{n,k} := \binom{n}{k}^2 \binom{n+k}{k}^2$

And we find a common recurrence relation for:

- $a_n := \sum_{k=0}^{n} c_{n,k}$
- $b_n := a_n z_n + \sum_{k=1}^{n} \sum_{m=1}^{k} \frac{(-1)^{m+1}}{2m^3 \binom{n}{m}\binom{n+m}{m}} c_{n,k}$

And the difficult part

- In general, it is a difficult problem to invent a recurrence relation for a given sequence.
- In general, it is even difficult to prove that a given sequence satisfy a recurrence relation.

P-recursive sequences

A sequence $u := (u_n)_{n \in \mathbb{N}} \in \mathbb{K}^{\mathbb{N}}$ is called P-recursive if it is solution of a linear recurrence relation with coefficients in $\mathbb{K}[n]$.

Examples: $F_{n+2} = F_{n+1} + F_n, \quad nu_{n+2} - (n^2 + 100)u_{n+1} - u_n = 0$

When the recurrence has order 1 (and $u_0 \neq 0$) u is called a hypergeometric sequence.

Example: $(n+1)u_{n+1} = nu_n$

An elementary remark

As a finite linear relation between a sequence and its shifts, P-recurrence relations impose a structure of finite dimensional vector space to their set of solutions.

Hence to establish the equality of two P-recursive sequences, we can:

- Find a common equation annihilating both sequences
- Verify that the two sequences are point-wise equal on a sufficient set of initial conditions.

Notations

We first define:

- a shift operator $S_n : (u_n) \mapsto (u_{n+1})$.
- a multiplication operator $n\cdot\ (u_n) \mapsto (nu_n)$

Now a linear recurrence relation with polynomial coefficients can be described as a (non commutative) polynomial:

$$(n+3)u_{n+2} + n^2 u_{n+1} + 3u_n = 0$$

becomes:

$$P(u) = 0 \quad \text{with } P := (n+3)S_n^2 + n^2 S_n + 3$$

If a sequence involves more than one index, like in (u_{nk}), we denote $S_k, k\cdot, \dots$ the analogue operators relative to the indices $k, \dots$.

Closures

Implementation in computer algebra systems

The algorithms described on the schema (and variants, and extensions) are implemented in computer algebra systems.
They can be use to discover and validate the recurrence relations invented by Apéry.

See for instance the Algolib Maple package
`http://algo.inria.fr/libraries/`

Example

- $z_n := \sum_{k=1}^{n} \frac{1}{k^3}$
- $c_{n,k} := \binom{n}{k}^2 \binom{n+k}{k}^2$
- $d_{n,k} := \frac{(-1)^{m+1}}{2m^3\binom{n}{m}\binom{n+m}{m}}$

We can find a common recurrence relation for:

- $a_n := \sum_{k=0}^{n} c_{n,k}$
- $b_n := a_n z_n + \sum_{k=1}^{n} \sum_{m=1}^{k} d_{n,k} c_{n,k}$

by successive bottom-up discovery of recurrence relations.

See `http://algo.inria.fr/libraries/autocomb/` for an "Algolib aided version of Apéry's proof".

Toward a formal proof of correctness, in Coq

In order to validate the output of these computer algebra algorithms with a formal proof of correctness we can:

- Use the Coq proof assistant as a programming language to implement the algorithm, and prove formally its full correctness
- Use an external oracle (Maple) to perform the desired calculation and prove formally properties of the object output without any assumption on how it has been computed.

Sharing the efforts

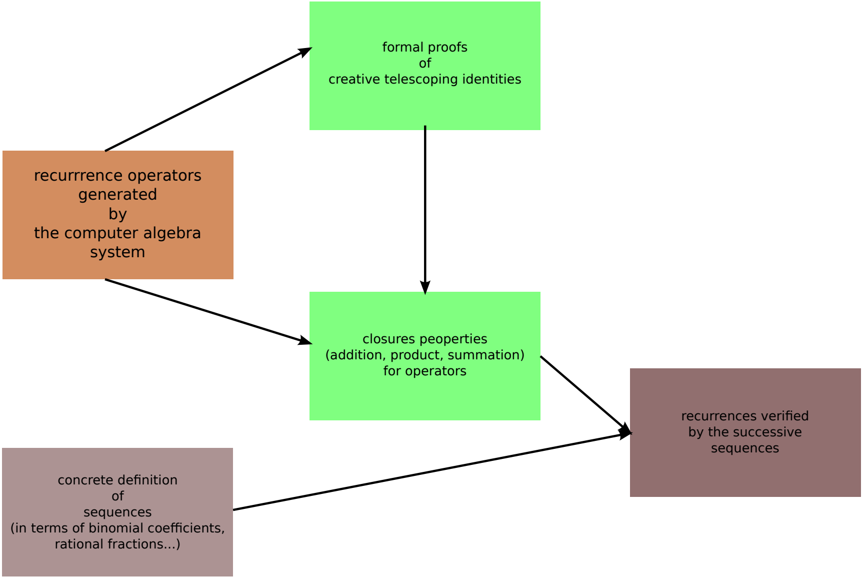
In our context, we can benefit from the external computation, that we will only verify (and not produce) using Coq:

- The computer algebra system explores the search space, and produces recurrence operators;
- The proof assistant validates the correctness of the operator by applying it to the sequence and normalizing the obtained expression to zero.

We hence need efficient formal-proof producing normalization algorithms implemented in Coq.

Example

Organization of the formal Coq development



formal proofs
of
creative telescoping identities

recurrrence operators
generated
by
the computer algebra
system

closures peoperties
(addition, product, summation)
for operators

recurrences verified
by the successive
sequences

concrete definition
of
sequences
(in terms of binomial coefficients,
rational fractions...)

A Maple session in Coq?

The interaction with the Coq system is for now far from the interaction mode a computer algebra offers to its users:

- The choice of datastructure is delicate;
- Computations are difficult to control;
- Little support is offered by the system to explore and manipulate (rather) large expressions generated by the computer algebra system.

Computer algebra issues

The is a discrepancy between the algebraic objects the computer algebra system manipulates and the actual total operators we need for proving properties on concrete instances.

Current state of the formalization

- Elementary number theory (integrality of some rational numbers)

  Done.

- Validation of recurrence relations

  Almost done

- Asymptotic study of $(a_n\zeta(3) - b_n)$

  Not yet started

- Asymptotic behaviour of $\mathrm{lcm}(1, \ldots, n)$

  May be difficult (cf. Prime Number Theorem)

Perspectives

This algebraic abstraction and its associated elimination theory also apply to the differential analogue of LDE with polynomial coefficients.

- Tools for the automated and certified validation of identities;
- Standardized libraries for mathematical functions in Coq;
- Interaction with other certification efforts on numerical issues (see Ioana's talk this afternoon).