Computer proofs with high-level concepts

Koen Vervloesem, Katholieke Universiteit Leuven
<koen.vervloesem@student.kuleuven.be>

Tuesday 27 March 2007

Perspectives on Mathematical Practices 2007

Computer proofs with high-level concepts

- Introduction
- Mathematical concepts and proofs
- Mathematical concepts in computer proofs
- Computation and reasoning
- Conclusion

Introduction

Computer proofs and human proofs

- **Mathematical practice: mathematics as a human activity. What with computers proving theorems?**
- **Observation: there is a difference between understanding a computer(-assisted) proof and a “human” proof.**
- **Explanation: we have to look at the properties of proof**
  1. **Structure**
  2. **Size**
  3. ***Concepts***
- **Claim: an important difference is to be found in the level of the concepts used in computer proofs and human proofs.**

Concepts

- **I show a (typical) example of a simple theorem and two different proofs of it. Mathematicians prefer the more “conceptual”.**
- **I give some examples of computer proofs and the mathematical concepts they use. They score poorly on the conceptual level.**
- **Concepts are connected to the dichotomy computation/ reasoning in proofs.**
- **This explains:**
  1. **why people object to computer proofs because of a lack of understanding;**
  2. **why computers cannot prove some simple results.**

Mathematical concepts and proofs

Cassini's identity for Fibonacci numbers

- **Fibonacci numbers:**
  - $F_0 = 0$
  - $F_1 = 1$
  - $F_n = F_{n-1} + F_{n-2}$ **for n > 1**
- **Cassini's identity:**

$$F_{n+1}F_{n-1} - F_n^2 = (-1)^n \text{ for } n \geq 1$$

Proof 1: by induction and calculation

- For n = 1: $1.0 - 1^2 = (-1)^1$
- Supposing the theorem holds for n = m, then:

$$F_{m+2}F_m - F_{m+1}^2 = (F_{m+1} + F_m)F_m - (F_m + F_{m-1})^2$$
$$= F_{m+1}F_m + F_m^2 - F_m^2 - 2F_mF_{m-1} - F_{m-1}^2$$
$$= F_{m+1}F_m - 2F_mF_{m-1} - F_{m-1}^2$$
$$= (F_m + F_{m-1})F_m - 2F_mF_{m-1} - F_{m-1}^2$$
$$= F_m^2 + F_{m-1}F_m - 2F_mF_{m-1} - F_{m-1}^2$$
$$= F_m^2 - F_mF_{m-1} - F_{m-1}^2$$
$$= F_m^2 - (F_m + F_{m-1})F_{m-1}$$
$$= F_m^2 - F_{m+1}F_{m-1} = -(-1)^m = (-1)^{m+1}$$

- By the induction hypothesis, this concludes our proof.

Proof 1: by induction and calculation

- **The proof uses a fairly limited set of concepts: the definition of Fibonacci numbers, some algebraic calculations and induction.**
- **The proof verifies the theorem, but doesn't give us understanding: why does the identity hold?**

Proof 2: by matrix theory

- **Key insight: left-hand side of Cassini's identity can be interpreted as the determinant of a 2 by 2 matrix of Fibonacci numbers.**
- **Then it is a one-liner:**

$$F_{n-1}F_{n+1}-F_n^2 = \det\begin{bmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{bmatrix} = \det\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^n = \left(\det\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}\right)^n = (-1)^n.$$

- **This proof shifts the level of concepts to another domain: from integers to matrices.**
- **Calculations: trivial.**
- **You can see *why* the identity holds. This proof *explains* why the right-hand side is $(-1)^n$.**

Why this example?

Many computer proofs have access to a limited set of concepts which they use to calculate.

- **The proofs look like our first proof by calculation.**
- **People don't like them because they give no insight.**

Jeremy Avigad gives an analysis of other examples of different proofs of a theorem and their virtues in *Mathematical Method and Proof* (2006).

- **$2^{32} + 1$ is not a Fermat prime.**
- **Products of sums of squares.**

Mathematical concepts in computer proofs

Example 1: four-colour theorem

Theorem: any planar map can be coloured using four colours in such a way that regions sharing a common boundary, other than a single point, do not have the same color.

Proof:

- **Appel, Haken, Koch (1977)**
- **Robertson, Sanders, Seymour, Thomas (1996)**
- **Gonthier, Werner (2004)**

Evolution in the proofs: from computer-assisted to completely computer-verified (in Coq)

Gonthier's formalization

Gonthier had to formalize intuitive graph-theoretic concepts in combinatorial properties.

- **e.g. formalization of plane graph using the Jordan curve theorem:**

“However, this approach results in proofs that, while convincing for humans, are difficult to formalize because they contain an informal mix of combinatorics and topology.”

—

Georges Gonthier, *A computer-checked proof of the four colour theorem* (2004), p. 5

Formalization versus intuition

Gonthier's alternative: hypermaps

Problem: “gap between the intuitive, picture-rich proof outline, and the very precise logical statement that had to be fed to the Coq proof assisant”

Problem: the more formalized the proof, the less intuitive it is.

Example 2: Robbins problem

Robbins algebra:

1. **Commutativity: x + y = y + x**
2. **Associativity: (x + y) + z = x + (y + z)**
3. **Robbins axiom: ~(~(x + y) + ~(x + ~y)) = x**

Theorem: each Robbins algebra is a Boolean algebra

Proof: by EQP

- **Long formulas, many parentheses, 12 steps.**
- **Problem with this proof: it's not *conceptual*, EQP gets some equations and starts to reason with them, but it doesn't invent new concepts.**

An anthropomorphized proof

Stanley Burris, *An anthropomorphized version of Mc-Cune's machine proof that Robbins algebras are Boolean algebras* (1996)

- **Longer proof, auxiliary variables, e.g. T := D + E + y + y.**
- **4 pages:**
  - **1 page of definitions and auxiliary variables**
  - **3 pages of proof steps.**
- **105 short equations, easily checked by hand.**

XCB-problem

Proof: by Otter (Wos, Ulrich, Fitelson)

“That such assistance was invaluable, and perhaps indispensable, will occur to the reader who attempts to carry out by hand the condensed detachment of line 16 from line 12 to obtain line 17 in the proof given in the following section. The substitution instances of 12 and of 16 required for that condensed detachment are, respectively, 2,939 and 2,919 symbols in length, a consideration that may explain in part why these two questions about XCB remained unanswered for so long.”

—

***XCB, the last of the shortest single axioms for the classical equivalential calculus* (2003)**

Computation and reasoning

Forwards and backwards

“Typically computational steps move ‘forwards’ (from the known facts further facts are derived) and logical steps move ‘backwards’ (from the goal towards the hypothesis, as in it would suffice to prove). The mixture of logic and computation gives mathematics a rich structure that has not yet been captured, either in the formal systems of logic, or in computer programs.”

—

Michael Beeson, *Automatic derivation of epsilon-delta proofs of continuity* (1998)

The two activities of proof

Proof = computation + reasoning

- **Proofs of automated theorem provers: logical steps.**
- **Proofs of the four-colour theorem: mainly computations.**

Mix of these two: Beeson's program Weierstrass. It proved the irrationality of e.

- **The proof is understandable and relatively short.**
- **The proof looks like a human proof.**

Concepts in computation and reasoning

Computation:

- **Concepts on the same level / in the same domain.**
- **Goes (quasi-)automatically (e.g. proof 1 of Cassini's identity).**

Reasoning:

- **Relate concepts (from different domains).**
- **Requires insight / trying (e.g. proof 2 of Cassinni's identity).**

A good human/computer proof needs computation *and* reasoning.

Conclusion

Conclusion

Looking for concepts

“What mathematicians are largely looking for from each other's proofs are new concepts, techniques and interpretations. Computer proofs certainly give information concerning the truth of a result, but very little beyond this.”

—

David Corfield, *Towards a philosophy of real mathematics* (2003), p. 56

Principia Mathematica and computer proofs

“One of the reasons why the Principia are so rarely read is that the main ideas of the proofs are no longer visible in very long and very detailed proofs. ”

—

Manfred Kerber, Martin Pollet, *On the design of mathematical concepts* (2002)

- **The concepts used are not adapted for understanding.**
- **Computer proofs and Principia Mathematica are mostly interested in truth.**

Mixed concepts

Lagrange's theorem: for any finite group G, the order of every subgroup H of G divides the order of G.

- **Concepts used in the theorem: subgroups, mappings, numbers.**
- **Simple and short proof.**
- **No theorem-proving program can find a proof now.**

Tony Huang's Master's thesis (2002): 150 exercises in ring theory, only 14 of them could be formulated in first order ring theory. Otter can prove these 14.

Conclusion

- **Many human proofs:**
  - **contain a mix of concepts of different domains, and this gives insight;**
  - **combine computation and reasoning.**
- **Many computer proofs:**
  - **use a limited set of concepts, and hence have not much possibilities to give insight;**
  - **use computation *or* reasoning.**
- **If computers can use richer sets of concepts:**
  - **people understand the proofs better;**
  - **computers could prove more results;**
  - **computer provers could become part of mathematical practice.**

Conclusion (Continued)