Stam's Identities Collection: A Case Study for Math Knowledge Bases

Bruno Buchberger(✉)

Research Institute for Symbolic Computation (RISC),
Johannes Kepler University, Linz, Austria
bruno.buchberger@risc.jku.at
http://www.risc.jku.at/people/buchberg/

Abstract. In the frame of the work of the Working Group "Global Digital Mathematical Library", Jim Pitman proposed Aart Stam's collection of combinatorial identities as a benchmark for "digitizing" mathematical knowledge. This collection seems to be a challenge for "digitization" because of its size (1300 pages in a .pdf file) and because of the fact that, for the most part, it is hand-written. However, after an in-depth analysis, it turns out that the real challenges are of mathematical and logical nature. In this talk we discuss what digitization of such a piece of mathematics means and report on various tools that may help in this endeavor. The tools range from technical tools for typing formulae all the way to sophisticated algebraic and reasoning algorithms. The experiments for applying these tools to Stam's collection are currently carried out by two of the working groups at RISC.

1 The Problem

Aart Stam's collection of combinatorial identities (Stam 2012) consists of hundreds of identities that show how formal sums involving bionomial coefficients can be simplified. The collection also explains how these identies can be proved using various proof techniques.

In the context of the "Global Digital Math Library" project, Jim Pitman (Pitman 2015) proposed to consider this collection as a benchmark for the "digitization" of mathematical knowledge. We are faced with the challenge how the extremely valuable knowledge contained in such a collection can be transformed to a form in which the individual identities can be stored, accessed, and processed by algorithmic tools over the web. One might think that the task should and could be decomposed into a first step of (automated) translation of the hand-written formulae into LaTeX or any other mathematical expression format and a second step of processing the LaTeX formulae by sophisticated algorithms and tools from computer algebra and automated mathematical reasoning. However, given current technologies, we will show that this may not be the most reasonable approach. In fact, we will see that it already can be questioned whether the individual identities need to be stored or, alternatively, may be generated and / or proved on demand!

G.-M. Greuel et al. (Eds.): ICMS 2016, LNCS 9725, pp. 437–442, 2016.
DOI: 10.1007/978-3-319-42432-3_55

2 The RISC Approach

For looking into the feasibility of formalizing Stam's paper, we installed a seminar at our RISC institute (my Theorema group and Peter Paule's Symbolic Combinatorics group) for working together on the formalization of Stam's paper. Our team consists of:

- Theorema Group: Bruno Buchberger, Alexander Maletzky, Wolfgang Windsteiger.
- From the Symbolic Combinatorics Group: Peter Paule, Christoph Koutschan, Clemens Raab, Silviu Radu, Carsten Schneider.

After in-depth discussion, we came up with the following decomposition of work and distinction between the various aspects of the problem:

a. *Translating the formulae into predicate logic form (or any variation of this form), but still in the usual nice two-dimensional appearance used in math papers and typing the formula in this format:*
This can be easily done in Theorema. I did some timing experiments and my estimate is that, for the approximately 1200 formulae in the section "Tables" (the kernel of Stam's paper) I would need approximately 60 h. After this, all formulae would be available in correct logical form that could be translated to any other logic form automatically. Also, after the formalization, hyperlinks to all formulae would be available. Here is a view to the first few formulae in Theorema notation (which could be changed according to the taste of users):

$\forall n \in \mathbb{N}$

From D(19):

$$\left(\sum_{k=0,\ldots,n} \binom{n}{k}\right) == 2^n \tag{1}$$

From D(19):

$$n \geq 1 \Rightarrow \left(\left(\sum_{k=0,\ldots,n} (-1)^k \binom{n}{k}\right) == 0\right) \tag{2}$$

$$\left(\left(\sum_{k=0,\ldots,n} (-1)^k \binom{x}{k}\right) == (-1)^n \binom{x-1}{n}\right) \tag{3}$$

The important thing is that, internally, the complete parse tree of the formulae as a *Mathematica* nested expression is available. Thus, automated translation to any other formalization, to the input format of arbitrary reasoners, and of course also to pretty-print LaTeX, would be possible. This is in clear distinction to formulae presented, first, in LaTeX, which does not display the logical structure of the formulae and from where automated translation to formulae in logic is *not* possible.

For example, the *Mathematica* formula:

$$\sum_{i=1}^{n} k^2$$

has the internal representation

Sum[Power[k,2], List[i,1,n]]

which reveals the structure completely. Theorema formulae, internally are also *Mathematica* nested expressions but with a structure that is closer to some common forms of predicate logic.

b. *Automated proofs of all formulae using the "old-fashioned" proof methods:* Some of these methods (simplification, induction, summation quantifier inference rules as described in (Buchberger 1980)), are already implemented in Theorema. Some adjustments are necessary though. Stam lists approximately 15 "old-fashioned" proof methods. I estimate that we can implement all of them in Theorema with an effort of about one person year. Most of them, however, are superseded by the "modern" proof methodology, see remark c., or already have some flavor of the "new-fashioned" proof methods, e.g. Ergoychev's method. See however also remark g.
   - reduction to known formulae
   - rearranging factorials
   - Fibonacci
   - Lucas
   - poly of convolution type
   - specialization in general summation formulae
   - the complex argument
   - induction over naturals
   - recurrence
   - finite differences
   - Newton interpolation
   - inverse relations (to do with convolution)
   - inclusion - exclusion
   - multisection of sums
   - expansion of factor in the summand
   - the beta integral
   - generating functions
   - partial fractions
   - Egorychev' s method.

c. *Automated proofs of the formulae using "new-fashioned" proof methods, I call them "Algebraic Simplification Methods":*
   These methods proceed by translating the formulae into objects in suitable algebraic (polynomial) domains and sophisticated new simplification techniques based on new math results for these poly domains (e.g. the noncommutative Gröbner bases methodology). This is prominent research activity of about 15 people in the world over the past 20 years. In this talk and

extended abstract, I cannot give a fair account of the individual contributions of the key players in this research field. Ground was laid by Doron Zeilberger (the "holonomic systems approach to special function identities") together with Herb Wilf, George Andrews and Marko Petkovsek. Today the methods (with software implementation) by Peter Paule and his former PhD students Manuel Kauers, Christoph Kouschan, Carsten Schneider et al., and by Frederic Chyzak seem to be the most advanced. The literature on the field is contained in the recent monograph (Kauers 2011).

After detailed inspection of all the formulae in Stam's paper, we are pretty sure that 95% or more of these formulae can be proved completely automatically with the methods available in Paule's group. After having typed all formulae, the actual proof (verification) of all these identities, in typical cases, is a matter of a few seconds per formula.

d. *Formal, and maybe automated, proof of the correctness of the algebraic theory (like Gröbner bases etc.) which is behind the methods in c.:*
This is a major task, which goes far beyond Stam's paper but would of course be an essential and interesting part of a future comprehensive paper on combinatorial identities. For the commutative case of Gröbner bases theory, I am working on this with one of my PhD students, see (Maletzky 2016) and, in fact, this theory is now completely formalized *and* formally proved. This includes formalization and formal proof of my algorithm for computing Gröbner bases within the same logic in which the formalization of the rest of the theory is done. In fact, the execution of the algorithm on concrete input is also done within the same logic. Many more theses etc. would be necessary for formalizing and formally proving all current theory behind symbolic combinatorics. I consider research of this type as the essential goal of future formal math. Thinking further ahead, the question arises if, with today's mathematical knowledge and methodology of type c., it would at all make sense to type all the formulae in Stam's paper. My answer comes in the next items:

e. *Automated generation of combinatorial identities:*
One could write a "conjecture generator" that automatically generates all (and more of) the formulae listed in Stam's paper as conjectures and then submits the conjectures to the methods in c., keeping those that are true. I have ideas for this and did this already on a smaller scale for a different area.

f. *Proof of identities on demand:*
Alternatively, one just would not any more generate tables of identities but would "wait for the user" who has a particular instance of any of the formulae in the table and wants to get an automated verification by the methods in c. Even more attractively, by the methods in c., one can obtain automatically a simplified right-hand side if one provides a complicated left-hand side. This is similar to the situation in symbolic integration: We do not any more store integral tables in math systems like Mathematica, Maple etc. but, rather, one uses Risch' algorithm (Risch 1969) or extensions thereof for generating the integrals on demand.

g. *Providing "old-fashioned" proof methods in the presence of "modern" proof methods:*
My personal view on the question of "old-fashioned" proof methods versus "modern" proof methods is as follows: The ("manual" or automated) proof of formulae by some "older" proof techniques needs extra "handcrafting" for each individual formula. Often (95% or more), proofs of these formulae can be obtained, without extra hand-crafting, completely automatically by a "newer" proof technique. However, still, there may be reasons why a mathematician working in a particular area, as for example statistics or algorithm complexity, may want to see proofs generated by newer *and* older methods. Also, he may want to see "complete tables of identities" (like Stam's collection) even if they would not be necessary any more in the presence of newer methods. The reason for this may be:
- Proofs generated by various different methods may give various different insights about the formulae proved.
- The use of older or newer proof techniques and the desire of seeing "tables" of formulae may depend on the particular application of the identities in other fields of mathematics (Pitman 2015).
- The relation between "older" and "newer" proof techniques is, in fact, as old as mathematics. However, so far, in the lifetime of a mathematician, the proof techniques in his field did not really change. Logically, proceeding from "older" to "newer" proof techniques is an important ingredient of mathematics. We pointed this out, for example, in (Buchberger 2012). In essence, the transition from "old" to "new" is the transition from the "object" to the "meta" level of mathematics. Therefore, we advocate that modern math proving systems have to provide means for proceeding from the object to the meta level (e.g. level b. to d. and then to the application of d. to c.). In Theorema, this is an important design principle and we showed its feasibility in the work on formalizing Gröbner bases theory.

In the talk, I will report on the current state of the joint work of our team on the various aspects above.

Acknowledgements. The work described in this talk is carried out in a team at RISC consisting of B. Buchberger, C. Koutschan, A. Maletzky, P. Paule, C. Raab, S. Radu, C. Schneider, and W. Windsteiger.

References

Buchberger, B., Lichtenberger, F.: Mathematics for Computer Scientists. Springer, Heidelberg (1980). (in German)

Buchberger, B., Mathe is meta. In: Invited Talk at the Summer School "Summation, Integration and Special Functions in Quantum Field Theory", 9–13 July. RISC. Johannes Kepler University, Castle of Hagenberg, Austria (2012)

Kauers, M., Paule, P.: The Concrete Tetrahedron. Texts and Monographs in Symbolic Computation. Springer, Vienna (2011)

Maletzky, A.: Formalization of Gröbner Bases Theory in Theorema (working title). Ph.D. thesis, July 2016, to appear

Pitman, J.: Personal Communcation to the Working Group "Global Digital Math Library", December 2015

Risch, R.H.: The problem of integration in finite terms. Trans. Am. Math. Soc. **139**, 167–189 (1969)

Stam, A.: Binomial Identities with Old-fashioned Proofs, Manuscript, University of Groningen (2012)