Proceedings of the 4th Conference of Mathematical Society of Moldova
CMSM4'2017, June 28-July 2, 2017, Chisinau, Republic of Moldova

Algebraic Methods in Automated Reasoning

Tudor Jebelean

Abstract

We present succintly a case study on automatic generation of natural–style proofs in elementary analysis, by employing algorithms from computer algebra. In order to produce proofs which are similar to those realised by human mathematicians, we use a system similar to sequent calculus, in which the most difficult steps consist in finding the witness terms for the existential goals and the instatiation terms for the universal assumptions. We study how these can be found by using computer algebra algorithms, and what are the current limitations and perspectives of this approach.

Keywords: computer algebra, natural–style proving.

1 Introduction

The production of natural–style proofs (that is: proofs which are similar to the ones written by human mathematicians) may be of increasing importance in the future, because understanding proofs may become crucial in order to trust them, or to guide the difficult steps, or to use them in tutorial presentations. The *Theorema* system [1] aims at constructing such proofs in various areas of mathematics.

For proofs in elementary analysis, in which many notions are defined using complex formulae with alternating quantifiers, we developed the original strategy of *S-decomposition* [2], which is particularly suitable for treating such formulae. In such proofs, the tasks which are most difficult to automate consist in finding the witness terms for the

existential goals and the instatiation terms for the universal assumptions. Our case study in *Theorema* demonstrates how these tasks can be partially solved by using cylindrical algebraic decomposition and quantifier elimination [3]. Although in linear cases this approach is mostly successful, in problems of higher degree it often fails. We investigate various methods to improve the performance in these cases.

2 Natural–Style Proving

In the *Theorema* system we aim at producing proofs which are similar to those realised by humans[1]. For instance, let us consider the proof of the statement: *"The sum of two convergent sequences is convergent"*. The convergence of a sequence f (function from naturals to reals) is defined by the following formula with alternating quantifiers (ϵ is real, m, n are naturals):

$$\exists_{a} \; \forall_{\epsilon>0} \; \exists_{m} \; \forall_{n\geq m} \; |f(n) - a| < \epsilon$$

The proof shows that the instance of this formula for $f_1 + f_2$ (the *goal*) is implied by the two instances of the same formula for f_1 and for f_2 the (*assumptions*).

The natural–style proof proceeds by eliminating in parallel the same quantifiers from these three formulae, as described in [2]: In the existential assumptions, the quantified variable is transformed into the so called "such a" Skolem constant, and after that the existential goal is proved by using the appropriate "witness term". In the universal goal, the variable is transformed into the so called "arbitrary but fixed" Skolem constant, and after that the universal assumptions are instantiated with the appropriate terms. A special feature of our approach is to treat separately the condition associated to the quantified variable (in the formula above: $\epsilon > 0$ and $n \geq m$), which generate separate independent goals.

[1] This is not the same as *natural–deduction*.

3 Using Algebraic Methods

While the Skolemization steps mentioned above have a trivial implementation, the construction of the witness terms and of the instantiation terms is quite difficult to perform automatically, because the necessary information becomes available only later in the proof. We experimented the use of algebraic techniques for finding these terms, following a method presented in [4]. In the example above, the successive steps of the proof are essentially equivalent to a prenex decomposition of the whole original implication, and formulae obtained are:

$$\underset{m,n}{\forall}\ \underset{p}{\exists}\ \underset{q}{\forall}\ (q \geq p \implies q \geq m \wedge q \geq n)$$

$$\underset{a_1,a_2}{\forall}\ \underset{a}{\exists}\ \underset{\epsilon}{\forall}\ \underset{\epsilon_1,\epsilon_2}{\exists}\ \underset{x_1,x_2}{\forall}\ (|x_1-a_1| < \epsilon_1 \wedge |x_2-a_2| < \epsilon_2 \implies |(x_1+x_2)-a| < \epsilon)$$

For proving the first formula we can use CAD–based quantifier elimination (QE), and the answer is **true**, but this does not reveal a natural–style proof. If we use QE on the same formula without $\underset{m,n}{\forall}\ \underset{p}{\exists}$, then we obtain a relation between m, n, p which allows to infer the expression for p (will be the maximum of m and n) by adequate post-processing. For proving the second formula, one can apply QE/CAD first on the formula without $\underset{a_1,a_2}{\forall}\ \underset{a}{\exists}$, which returns $a = a_1 + a_2$. Then one substitutes a and eliminates further the quantifiers $\underset{\epsilon}{\forall}\ \underset{\epsilon_1,\epsilon_2}{\exists}$, on which QE/CAD returns $\epsilon_1 + \epsilon_2 \leq \epsilon$, which allows to infer appropriate witnesses for ϵ_1 and ϵ_2, namely $\epsilon/2$.

The above approach is not very efficient, because it needs a repeated CAD for each formula. Therefore we are investigating possible adaptations of the algorithm which can extract all the necessary information in one pass. Moreover, while the algorithm works relatively fast for expressions of degree one (as above), it is overcoming the system resources for expressions of higher degree (for instance, when we try to do the analoguous proof for *product* instead of *sum*). For overcoming this problem we are operating various simplifications of the original formulae, which need less computation, but still are able to reveal the same desired terms.

4 Conclusion

The use of algebraic algorithms for producing specific terms in natural–style proofs is successful at least in simple cases, however for more complex problems it becomes unproductive. Performing various case studies in elementary analysis appears to hold the promise of finding more efficient and effective versions of the algorithms, which will be able to solve more complex problems.

Acknowledgments. Supported by the project "Satisfiability Checking and Symbolic Computation" (H2020-FETOPN-2015-CSA 712689).

References

[1] Buchberger, B., Jebelean, T., Kutsia, T., Maletzky, A., Windsteiger, W.: *Theorema 2.0: Computer-Assisted Natural-Style Mathematics*. JFR 9(1), 149–185 (2016)

[2] Jebelean, T., Buchberger, B., Kutsia, T., Popov, N., Schreiner, W., Windsteiger, W.: *Automated Reasoning*. In: Buchberger, B. et al. (ed.) Hagenberg Research, pp. 63–101. Springer (2009)

[3] Collins, G.E.: *Quantier elimination for real closed fields by cylindrical algebraic decomposition*. In: Automata Theory and Formal Languages. LNCS, vol. 33, pp. 134–183. Springer (1975)

[4] Vajda, R., Jebelean, T., Buchberger, B.: *Combining Logical and Algebraic Techniques for Natural Style Proving in Elementary Analysis*. Mathematics and Computers in Simulation 79(8), pp. 2310–2316 (April 2009)

Tudor Jebelean

RISC–Linz, Johannes Kepler University
Email: `Tudor.Jebelean@jku.at`