Adapting Cylindrical Algebraic Decomposition for Proof Specific Tasks

Erika Abraham[1] and Tudor Jebelean[2]

[1] RWTH Aachen University, Germany
[2] RISC-Linz, Johannes Kepler University, Austria

Keywords: Cylindrical Algebraic Decomposition, Automated Reasoning, Satisfiability Modulo Theories

MSC: 68T15, 68W30

Cylindrical Algebraic Decomposition (CAD) [2] is a quite successfull, albeit expensive technique for quantifier elimination in the theory of real closed fields. Previous research [5] shows that this technique can be used for finding automatically witnesses in mechanical proofs for simple theorems in elementary analysis. Given a real-algebraic sentence whose truth we want to prove, intuitively, CAD as a quantifier elimination procedure is used to find for each existentially quantified variable a suitable term (witness) such that substituting that term for that variable leads to the satisfaction of the statement. Unfortunately, the approach [5] requires the invocation of the CAD procedure once for each existential quantifier block and can therefore be time–cost prohibitive, especially if the problems contain polynomials of higher degree.

The research reported here proposes improvements to the previous approach to increase the efficiency of finding witness terms. Instead of calling the CAD as a black box, our improvements are based on white-box CAD computations, where we make use of influencing the decomposition process and extracting CAD-internal information from the decomposition which cannot be extracted from the standard black-box output of the CAD (as implemented in, e.g., *Mathematica*). The main ideas are to (1) apply only partial CAD computations that are necessary for finding proper witness terms for the existentially quantified variables and to (2) reduce the effort to computing a single CAD and exploit its internal representation to construct witnesses for all existentially quantified variables in one step. Our technique can also be applied to strengthen Satisfiability Modulo Theories (SMT) solvers, originally designed for deciding the satisfiability of quantifier-free logical problems, with abilities to provide symbolic solutions for satisfiable problem instances even for quantified real-algebraic problems.

We illustrate our approach by an example. Consider the notion of *convergence of real sequences*, which is defined by the following definition with alternating quantifiers (f is a real function of natural argument, ϵ is real, m, n are naturals):

$$\mathrm{IsConvergent}(f) \;\Leftrightarrow\; \underset{a}{\exists}\ \underset{\epsilon>0}{\forall}\ \underset{m}{\exists}\ \underset{n\geq m}{\forall}\ |f(n)-a|<\epsilon$$

The proof of the statement "the sum of two convergent sequences is convergent" reduces (by a proof technique in natural style described in [4]) to the following two subgoals:

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

and

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

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 postprocessing. However, by using CAD in a specific way, we are able to extract the proper witness for p from the first QE process, by using the information provided by all the quantifiers, and also without the need to investigate all possible branches in the reconstruction phase: on the branches corresponding to the existential quantifiers, one only proceeds until a successfull instantiation is found.

For proving *the second formula*, in the current approach one has to 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. In the new approach, from the first application of the adapted CAD algorithm to the full formula we can obtain all the necessary witnesses, and we also avoid several instantiations on existential branches.

The same kind of problem in the case of product of sequences generates a QE/CAD process which is quite time consuming in the *Mathematica* implementation, and in fact a suitable simplification of the result is not possible. By applying our novel technique we aim to generate the necessary witnesses in a useful form and in a shorter time.

We implement the approach in the frame of the *Theorema* system [1] developed at RISC-Linz and in the frame of the `SMT-RAT` system developed at RWTH Aachen [3]. This also allows to compare the efficiency of the *Mathematica* implementation to the efficiency of a custom implementation, and to demonstrate the possibility of using external algebraic tools in *Theorema*.

Further work includes the developement of methods for the completion of natural style proofs after the appropriate witnesses are found, and the application of the proposed technique in further areas. For example, a relevant application would be to compute a symbolic description of the input-output behaviour of communicating processes.

Acknowledgments. Partially 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. 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)
3. Corzilius, F., Kremer, G., Junges, S., Schupp, S., Abraham, E.: `SMT-RAT`: An open source C++ toolbox for strategic and parallel SMT solving. In: Proc. of SAT'15. LNCS, vol. 9340, pp. 360–368. Springer (2015)
4. Jebelean, T., Buchberger, B., Kutsia, T., Popov, N., Schreiner, W., Windsteiger, W.: Automated Reasoning. In: et al., B.B. (ed.) Hagenberg Research, pp. 63–101. Springer (2009)
5. 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), 2310–2316 (April 2009)