An Open Environment for Doing Mathematics

Karsten Homann Jacques Calmet

Universität Karlsruhe
Institut für Algorithmen und Kognitive Systeme
Am Fasanengarten 5 · 76131 Karlsruhe · Germany
{homann,calmet}@ira.uka.de

Abstract

There are several possible approaches to integrate theorem provers (TP) and computer algebra systems (CAS). On one hand classical CAS usually offer a straightforward programming language with ad-hoc implementations of rewriting. One approach towards introducing theorem proving in CAS is an extension of Analytica [Clarke & Zhao 92], a Mathematica package to prove theorems in elementary analysis which solves an extensive collection of nontrivial mathematical problems.

On the other hand some classical TP were extended by techniques of symbolic computing, e.g. Otter allows to call external algorithms out of proofs. Specialized prover packages have been developed which are capable to perform symbolic mathematical computations. However, there are no environments integrating theorem provers and computer algebra systems which consistently provide the inference capabilities of the first and the powerful arithmetic of the latter systems.

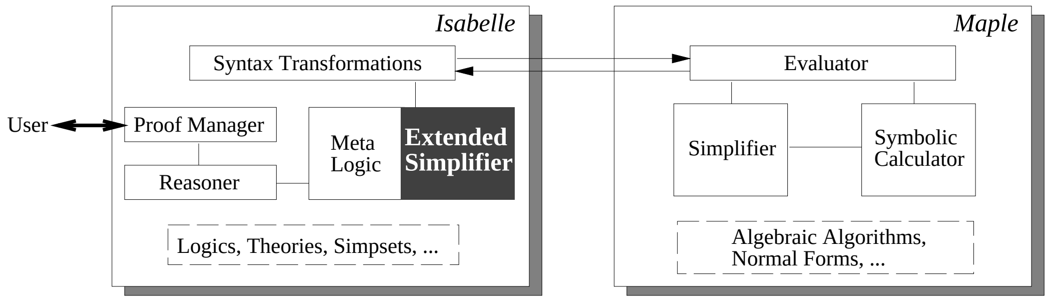
Another aspect is the integration of several systems in a common environment – an intelligent mathematical assistant. Different possibilities to integrate symbolic calculators and theorem provers are given in [Homann & Calmet 94]. The best possible mutual benefit is achieved either by a common knowledge representation or by standardized interfaces between all systems for doing mathematics. The former requires redesign of present CAS and TP or development of new systems. Such a development may lead to a common and explicit representation of the embedded mathematical knowledge, e.g. theorems, algorithms and types. Moreover, representing theorems corresponding to algebraic algorithms explicitly allows the systems to reflect and explain their behaviour instead of being black boxes. The design of the latter demands to develop a common and general communication language and appropriate interfaces.

Since it is doubtfull that one single system can satisfy the multitude and divergency of requests and problems the goal of this research is to a certain extend heterogeneous in the sense of integrating diverse systems. Ideally, such an *open mechanized mathematics environment* should be easily extendable and should provide interfaces to the existing widespread CAS and TP. The capability to connect such systems in a straightforward manner is the main step towards an open mechanized mathematics environment.

Some works exist to integrate different TP or CAS respectively. [Giunchiglia et al. 94] introduces an architecture for open mechanized reasoning systems which consists of a reasoning theory as well as a control and an interaction component. The long term goal is a methodology to construct complex systems as a composition of several reasoning systems. CAS/π [Kajler 92] is a powerful system-independent graphical user interface to most well-known CAS.

As an example for the development of a *common* environment for doing mathematics we implemented a prototype of an interface between the tactical theorem prover Isabelle [Paulson 94] and Maple. The interface is realized by extending the simplifier of Isabelle without any modification of Maple. Since we do not have to take into consideration any idiosyncrasies of Maple except its syntax,

it would be very easy to link Isabelle to another CAS as well. The simplifier is extended by the introduction of a new class of simplification rules called evaluation rules in order to make selected operations of Maple available. Additionaly, we specify syntax translations for the concrete syntax of Maple. They enable Isabelle to communicate with the computer algebra system to solve a set of exemplary problems. Another prototype connects Magma and the theorem prover Dtp to compute and prove theorems in group theory.



References

[Calmet & Campbell 94] J. Calmet, J.A. Campbell, *Artificial Intelligence and Symbolic Mathematical Computing,* Editors, Lecture Notes in Computer Science, 1994, to appear.

[Char et al. 92] B.W. Char, K.O. Geddes, G.H. Gonnet, B.L. Leong, M.B. Monagan, S.M. Watt, *Maple V Language Reference Manual,* Springer-Verlag, 1992.

[Clarke & Zhao 92] E. Clarke, X. Zhao, *Analytica – An Experiment in Combining Theorem Proving and Symbolic Computation,* Technical Report CMU-CS-92-147, Carnedie Mellon University, 1992.

[Giunchiglia et al. 94] F. Giunchiglia, P. Pecchiari, C. Talcott, *Reasoning Theories – Towards an Architecture for Open Mechanized Reasoning Systems,* 1994, to appear.

[Homann & Calmet 94] K. Homann, J. Calmet, *Combining Theorem Proving and Symbolic Mathematical Computing.* In [Calmet & Campbell 94].

[Kajler 92] N. Kajler, *CAS/PI: a Portable and Extensible Interface for Computer Algebra Systems.* In Proceedings of ISSAC-92, ACM press, 1992.

[Paulson 94] L.C. Paulson, *Isabelle — A Generic Theorem Prover,* Lecture Notes in Computer Science 828, Springer-Verlag, 1994.