Reasoning About the Reals: the marriage of HOL and Maple

John Harrison * & Laurent Théry **

University of Cambridge Computer Laboratory
Pembroke Street, Cambridge CB2 3QG, England. {jrh,tl}@cl.cam.ac.uk

Abstract. Computer algebra systems are extremely powerful and flexible, but often give results which require careful interpretation or are downright incorrect. By contrast, theorem provers are very reliable but lack the powerful specialized decision procedures and heuristics of computer algebra systems. In this paper we try to get the best of both worlds by careful exploitation of a link between a theorem prover and a computer algebra system.

1 Motivation

In the HOL theorem prover[5], a theory of real numbers has been developed, using a rigorous definition in terms of Dedekind cuts [8]. It is therefore possible to apply HOL to areas traditionally within the purview of Computer Algebra Systems (CASs). This offers two main benefits. Firstly, theorem provers are designed to manipulate proofs and theorems in a coherent and structured way, with all concepts clearly defined. By contrast, most CASs have no concept of 'logic' as such – they usually take an algebraic expression and return another purportedly 'equal' to it (perhaps not truly equal, but equal in some weaker sense, say, as meromorphic functions, e.g. $(x^2-1)/(x-1) = x+1$). Secondly, theorem provers are designed to give high standards of assurance, since their usual area of application is in the verification of critical systems, whereas CASs use various heuristic mechanisms which can lead to clearly false results, e.g. the evaluation of $\int_{-1}^{1} \sqrt{x^2}\, dx$ in Maple[3] returns 0.

To emulate the algorithms found in CASs would require a substantial programming effort, and the implementations would probably be much slower because of the need to justify every step. However in this paper we suggest that many problems can be solved quite easily, yet rigorously, by exploiting a link between the HOL theorem prover and the Maple CAS.

2 The connexion

Linking two systems such as HOL and Maple falls in the general framework of tools integration. In our particular example, the techniques developed in [2] have been applied in a straightforward manner. The organization involves three different processes: HOL, Maple and a bridge. The communication between them uses broadcasting messages. HOL and Maple send and receive messages with string values

* Supported by the SERC UK.
** Supported by the SERC UK and the DSTO Australia.

representing formulas in their own syntax. The role of the bridge is to coerce these values automatically.

There are two distinct ways in the prover to make use of the answers of the CAS. The first one is to trust the CAS: when given a request A, if the answer is B, the prover must accept the theorem $\vdash A = B$. Doing this violates the HOL consistency principle of deriving all theorems from the axioms. To limit this corruption, we can use a technique given in [7], namely to define in HOL a constant *Maple* which is logically equivalent to false. Then it is possible to produce theorems such as $\mathit{Maple} \vdash A = B$, because one can prove anything from falsity. This technique has two nice properties. Firstly it has a natural reading: providing Maple is correct, A equals B. Secondly it has a natural composition property: every theorem whose proof uses such a Maple theorem will automatically inherit the assumption *Maple*. Even if we have introduced inconsistency we can still recognize 'dubious' theorems from 'true' theorems.

The second solution is to *always* check the answers. This solution keeps the integrity of the prover but can only be of some interest when finding a solution of a problem is computationally more complex than verifying it [1]. Then we can delegate the 'finding' part to the CAS and automate the 'checking' part in the prover. An example of such a situation is given in the next section.

3 An example: Integration

Integration is a good example of a situation where finding the answer can be very difficult, but checking it can be done in a fairly simple mechanical way by differentiating and simplifying the result to verify that it works. This however presupposes that the Fundamental Theorem of Calculus $\int_a^b f'(x)dx = f(b) - f(a)$ is true. Since we are now working in a proper logical framework we must define the notion of integral rigorously and prove the Fundamental Theorem.

One way, following Newton, is to *define* integration as finding an antiderivative, i.e. the inverse of differentiation. However this does not correspond very well to the intuitive association of an area under a curve: since derivatives are Darboux continuous, we would be unable to integrate functions which have step discontinuities. The usual mathematical definitions are based on methods derived by Riemann and Lebesgue among others [11]. However neither the Riemann nor Lebesgue integral obeys the Fundamental Theorem, for example the derivative of the function $x^2 sin(1/x^2)$ (and 0 at $x = 0$) is not Lebesgue integrable over $[0, 1]$.

We have instead chosen to define the integral in HOL following the Kurzweil-Henstock theory [9, 10]. This is simpler to define than the Lebesgue integral, and has the Lebesgue integral as a special case. Moreover, it obeys the Fundamental Theorem of Calculus which has been *proved* in HOL. Hence we are justified in evaluating integrals by letting Maple find an antiderivative.

To see how the combined system can be used, suppose we want to prove the following theorem about the logarithmic function:

$$\forall x.\, 1 \leq x \Rightarrow \int_1^x \ln y \; dy < x^2$$

Using Maple as an oracle to compute the value of the integral produces the two following subgoals:

$$\forall x.\, 1 \leq x \Rightarrow \int_1^x \ln y \; dy = x \ln x - x + 1 \qquad \forall x.\, 1 \leq x \Rightarrow x \ln x - x + 1 < x^2$$

The first subgoal checks the validity of the answer of Maple, the second one is the initial theorem where the integral has been replaced by the value returned by Maple. On the first subgoal, it is now possible to apply the Fundamental Theorem to get:

$$\forall x.\, 1 \leq x \Rightarrow \ln x = (x \ln x - x + 1)' \qquad \forall x.\, 1 \leq x \Rightarrow x \ln x - x + 1 < x^2$$

The new first subgoal can then be proved in a straightforward manner from the standard theorems about differentiation of sum and product, the second from the theorem $\forall x.\, 1 \leq x \Rightarrow \ln x < x$.

4 Conclusion and Future Work

In this paper, we have shown a generic approach to combining the heuristics of computer algebra systems with the rigour of theorem proving. Although we have explored only small examples so far, the result seems quite promising. We intend to develop this initial work in two directions. First, we want to tackle more realistic examples where the use of the CAS will be of some real advantage. Second we want to extend our connexion to allow several CASs to be connected to HOL via an auction mechanism as in [6]. From this, we hope to get the best features of different CASs available to do proofs.

References

1. R. J. Boulton, *On Efficiency in Theorem Provers which Fully Expand Proofs into Primitive Inferences*, Technical Report 248, University of Cambridge 1992.
2. D. Clement, *A Distributed Architecture for Programming Environment*, Proceedings of ACM SIGSOFT'90 SDE-4, Software Engineering Notes, Vol 15, no. 6, Irvine Ca, 1990.
3. B. W. Char, K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan and S. M. Watt. *A Tutorial Introduction to Maple V*, Springer-Verlag, 1992.
4. A. Church, *A Formulation of the Simple Theory of Types*, Journal of Symbolic Logic 5 (1940) pp. 56-68.
5. M. Gordon, T. Melham, *Introduction to HOL*, Cambridge University Press 1993.
6. N.Kajler. *Cas/Pi: A Portable and Extensible Interface for Computer Algebra Systems*, Proceedings of ISSAC'92, ACM 1992.
7. M. Gordon, Private Communication.
8. J. R. Harrison, *Constructing the Real Numbers in HOL*, Proceedings of the 1992 International Workshop on the HOL Theorem Proving System and its Applications, North-Holland 1993.
9. R. Henstock, *The General Theory of Integration*, Clarendon Press, Oxford 1991.
10. J. Kurzweil, *Generalized Ordinary Differential Equations and Continuous Dependence on a Parameter*, Czechoslovak Mathematics Journal 7(82), pp. 418-446, 1958.
11. I. N. Pesin, *Classical and modern integration theories*, Academic Press, New York 1970.

This article was processed using the LaTeX macro package with LLNCS style