waukee, 1996.

Proofs in Computational Algebra: An Interface between DTP and MAGMA

JACQUES CALMET KARSTEN HOMANN

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

http://iaks-www.ira.uka.de/iaks-calmet/index-e.html

Extended Abstract

While computational algebra could potentially play an important role in advanced mathematics and computer science, it has been said that progress is often hindered through a lack of knowledge and skills on the part of researchers. A researcher wishing to take fully advantage of computational algebra must assess different sophisticated packages and gain an understanding of their capabilities. Consequently, to integrate a vast amount of specialized software has been one of the key motivations in the MAGMA project.

The design of languages and environments to combine and integrate several heterogeneous systems has been initiated in many areas. For instance, the integration of theorem proving and symbolic mathematical computing has recently emerged from prototype extensions of single systems to the study of environments enabling interaction among distributed systems. An overview of recent well-known projects on such cooperations is given in the references cited in [1] and classified in [3]. Communication and cooperation mechanisms for logical and symbolic computation systems enable to study and solve new classes of problems and to perform efficient computation through cooperating specialized packages.

We designed and implemented an interface between DTP[1] [6] and MAGMA [2]. DTP is an automated theorem prover which is based on resolution. It is freely available and written in COMMON LISP. As MAGMA does not provide interfaces for interaction we implemented communication through standards named pipes. DTP acts as master providing the users interface and calling MAGMA.

To illustrate some features of the combination we give three simple examples of a classical exercise book in group theory [4]:

[1] Don Geddis' Theorem Prover

- *Find a group of order 32 with the smallest set of conjugate classes.*
  MAGMA includes a database of all such groups thus they do not have to be generated. It is easy to guess candidates but difficult to prove the minimality, e.g. the generalized group of quaternions[2]. The cooperation consists of the following steps:

  1. MAGMA provides by request of the prover a database and its cardinality;
  2. DTP retrieves all objects in the database;
  3. the cardinality of conjugate classes of every object is computed by MAGMA;
  4. DTP computes the minimum and determines the result.

- *Show that $W3$[3] is isomorphic to the direct product of A_5 and $\mathbb{Z}_2$.*
  A simple algorithm for constructing a mono-morphism of G_1 to G_2 if G_1 is isomorphic to a subgroup of G_2 has been implemented for any finite permutation groups G_1 and G_2 in DTP and MAGMA. $W3$ is transformed by the Todd-Coxeter algorithm `CosetAction` to a permutation group of order 120 and the isomorphism is automatically verified by considering the conjugate classes.

- *Find a minimal n such that the group of quaternions Q is a subgroup of S_n.*
  Since Q is of order 8 we know that $n \leq 8$ and $n \geq 4$ because 8 divides $n!$. Although very inefficient, MAGMA could be called to test all possible values for n by a simple algorithm. A better solution to this problem is automatically computed by the combination of DTP and MAGMA based upon reasoning with Sylow theorems by stepwise elimination of S_4, S_5, S_6 and S_7. For instance, the Dieder group D_4 of order 8 is a subgroup of S_4. Since D_4 is a 2-Sylow group and is not isomorphic to Q the second can not be a subgroup of S_4. The same holds for S_5 because D_4 is still 2-Sylow in S_5. To eliminate S_6 is more difficult because one has to generate a 2-Sylow group, determine the order of the elements and check that there are less elements of order 4 as in Q. S_7 can be eliminated by the other Sylow theorems.

The preliminary goal of this work is to demonstrate by examples the advantages of possible combinations of theorem provers and computer algebra systems. The problems illustrated in this abstract are automatically solved by integrating both paradigms. [1] illustrates an interface between an interactive tactical theorem prover which acts as master to a computer algebra system. The long term goal to design and implement environments for cooperative mathematical problem solving remains the topic of ongoing research.

Acknowledgment

We owe thanks to our student Andreas Döring for helpful discussions and for implementing the interface [5].

[2] Relations $B^{-1}ABA, B^2A^8$.

[3] $W3$ is a sporadic Coxeter group with relations $x_1^2, x_2^2, x_3^2, (x_1 * x_3)^2, (x_1 * x_2)^3, (x_2 * x_3)^5$.

References

[1] C. Ballarin, K. Homann, J. Calmet
Theorems and Algorithms: An Interface between Isabelle and Maple. In A.H.M. Levelt (Ed.), Proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC'95), pp. 150–157, ACM Press, 1995.

[2] W. Bosma, J. Cannon
Handbook of Magma Functions, Sydney, 1994.

[3] J. Calmet, K. Homann
Classification of Communication and Cooperation Mechanisms for Logical and Symbolic Computation Systems. To appear in Proceedings of First International Workshop Frontiers of Combining Systems (FroCoS'96), Kluwer Series on Applied Logic, 1996.

[4] J.D. Dixon
Problems in Group Theory, Dover Publishing, 1973.

[5] A. Döring
Kooperation eines Theorembeweisers und eines Computeralgebrasystems, (in German), project report, IAKS, Universität Karlsruhe, 1994.

[6] D. Geddis
The DTP Manual, Stanford University, 1994.