Automation for Interactive Proof
Final Report

Lawrence C. Paulson

1 Background/Context

The idea of supporting interactive provers using automatic ones is an old one. An important early effort is the KIV system [1], which has been integrated with 3TAP. Hurd has integrated HOL4 with Gandalf [3], while Bezem et al. have integrated Coq with Bliksem [2]. None of these integrations appear to be used any more, and indeed none of their automatic theorem provers appear to be undergoing development.

More successful have been integrations of interactive tools with specially-constructed automatic components. Hurd has written his own resolution prover, Metis, and integrated it with HOL4 [4]. Paulson wrote a tableau-style prover, blast, for integration with Isabelle [11].

Much of the prior work suffers from poor usability. Users have to collect relevant lemmas manually and often must transform problems into a suitable form for the automatic prover. Our project has aimed to eliminate the need for problem preparation, such as the removal of higher-order features; it has sought to use background processing, exploiting modern multi-core architectures; it has aimed to deliver its results as source-level proof scripts, so that expensive searches need not be repeated.

Closest to our conception is the Ωmega system [13]. However, while Ωmega is an experimental architecture designed to integrate many different types of reasoners, our project looks for techniques that could work for existing and well-established tools such as Isabelle and HOL4. Like the Ωmega group, we base the integration on open standards to avoid being tied to a single automatic prover.

2 Key Advances and Supporting Methodology

We have met all our main objectives, as stated in the Case for Support: to give interactive proof tools greatly improved automation; to develop the concept of an

interactive proof tool using background processing; to explore the formal relationships between first- and higher-order logic.

Integration. We have succeeded in integrating Isabelle with the automatic provers E, SPASS and Vampire. Other provers can easily be added to this list, particularly if they adhere to the TSTP format [14] for problems and solutions. Our integration uses background processing, though with the advent of multi-core machines, we have dismissed the idea of using remote processors. We have true "one-click" invocation: the system examines Isabelle's full lemma library and selects lemmas that appear relevant to the problem [9]. Higher-order problems are transformed into first-order ones using a novel and effective translation [6, 8]. This integration is both original and highly usable.

Proof reconstruction. Proof reconstruction is based on Hurd's Metis prover [5]. We have integrated the Metis prover with Isabelle, including proof reconstruction, following the existing HOL4 integration. Our system translates the output produced by E, SPASS or Vampire and generates one or more Metis calls that prove the required theorem. This proof script is presented to the user in source form [12].

Relevance filtering. Our work provides methods for selecting, from a huge lemma library, the few lemmas relevant to a given problem. Relevance filtering is necessary because automatic provers deliver poor results when given the standard lemma collection (consisting of a few hundred theorems) used with Isabelle's own automatic tools. We have developed and evaluated many strategies based on occurrences of constants in lemmas. Our methods work well enough that we can now use them with Isabelle's full lemma library of 7000 theorems [9]. By eliminating the need for user selection of lemmas, we get a more usable tool.

Higher-order translations. We have examined many approaches to translating higher-order problems to first-order logic. The complexity of Isabelle's type system precludes using an untyped translation as Hurd did, so the question is how much type information to retain. For the removal of λ-abstractions, we have compared combinators with λ-lifting. We have succeeded in finding a translation that delivers a good success rate [6, 8].

Experimental approach. One supporting methodology is systematic, extensive experimentation. This has allowed us to fine-tune many parameters and produce informative graphs. Fig. 1 is one example. The graph displays (on the vertical axis) the success rate for each of five translations as the runtime per problem increases from 20 to 300 seconds. Each data point represents attempts to prove 153

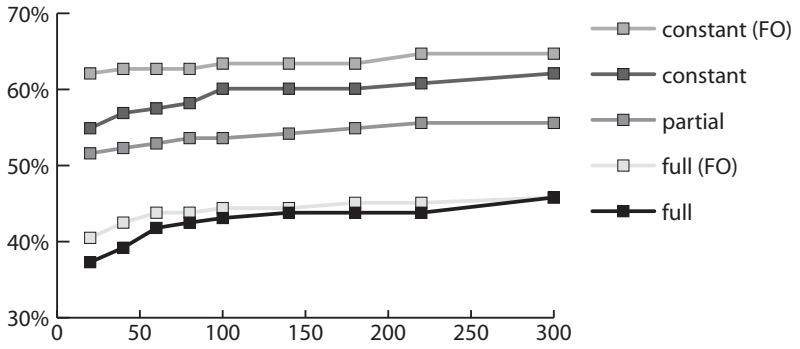

Figure 1: Success Rates With Various Translations

problems; its calculation could take 10 processor hours. Our relevance filtering experiments use a separate set of 285 problems. Such substantial experiments are seldom reported in the theorem-proving literature; the usual basis for comparison is a table reporting the outcome of single trials on a few dozen problems.

3 Project Plan Review

The actual research followed the proposed plan in most respects. All tasks were completed apart from Task 8: transferring the technology to HOL4. We could not undertake that task because our intended research assistant, Dr. Joe Hurd, instead took up a Fellowship at Oxford. Without Hurd's HOL4 expertise, it seemed more sensible to meet that task's objective of showing "that the techniques developed above are general, and not restricted to specific tools" by making our integration work with three different automatic provers. The HOL4 implementers should find our technology easy to transfer: we have documented our methods carefully, and HOL4 already has the Metis prover.

Dr. Claire Quigley was hired instead of Dr. Hurd. She served for 20 months before resigning due to ill health. She implemented the multi-tasking system [10], which spawns background processes to prove each of the current subgoals, returning the outcomes to the main process. She produced output in TPTP format for the provers E and Vampire, as well as DFG format for SPASS. She also implemented a first version of proof reconstruction for SPASS, but it proved to be fragile. Exactly emulating each of SPASS's many inference rules in Isabelle was difficult, especially as the output omits crucial details such as the specific subterm affected by a rewriting step. This suggested a new approach to proof reconstruction: to check each proof line by a separate call to the Metis prover. By performing a full search, Metis could easily cope with omitted information, literal reordering and other quirks of the proof it was given. This approach required integrating Metis

with Isabelle, which was a worthwhile goal in itself; it was undertaken by Dr. Kong Susanto, who took over as RA for the remaining 16 months. Hurd gave helpful advice on many occasions.

We decided to base proof reconstruction on E rather than SPASS. E outputs its proofs in TSTP format [14]; as other provers adopt this newly proposed standard, we should be able to support them. For provers that do not use TSTP, proof reconstruction consists of a single Metis call involving all lemmas used in the automatic proof. In this situation, Metis has to rediscover the entire proof in one step, but starting with tens of clauses rather than hundreds. Our experiments show that such calls succeed in over 90% of cases [12].

The project student, Jia Meng, successfully completed her PhD midway through the project. She obtained a post at NICTA, from which she continued to collaborate with us. She worked on two main tasks: relevance filtering and higher-order translations.

The full system, including line-by-line proof reconstruction, was working by the end of the project. Its incorporation into the Isabelle source repository was delayed by the arrival of Metis 2.0, which provided new interfaces and a revised treatment of proof reconstruction. The system is due for its public release as part of Isabelle 2007, which is planned for August 2007.

4 Research Impact and Benefits to Society

This project has yielded several publications [6, 7, 8, 9, 10, 12] as well as an implemented system. All of these items are in the public domain.

The practical benefits of the system are clear, especially to new users of Isabelle. We can expect the general idea to be copied: with today's multi-core architectures, proof tools should use background processing to analyse the user's problem.

Our work on higher-order translations is applicable to researchers building systems similar to ours. Our findings on relevance filtering are of broad importance to automated theorem proving. Our focus on huge problems has aroused much interest in the automated theorem proving community, which has formerly been preoccupied with difficult but compact problems. The new workshop *Empirically Successful Automated Reasoning in Large Theories* (with Paulson and Meng on the programme committee) is a sign of the importance now being attached to relevance issues.

5 Explanation of Expenditure

Staff costs were well underspent because the original award specified a salary point suitable to Dr. Hurd. The RAs actually employed were more junior.

Travel requirements can be difficult to predict. Both RAs were disinclined to travel, so this heading too was underspent.

It seemed appropriate to devote some of these unused funds to increasing the fileserver capacity, which was necessary in view of the large data sets used in the experiments. Consumables were therefore overspent.

6 Further Research and Dissemination Activities

There has been a Web page on this project since its inception,[1] and all project publications can be downloaded.[2] The full source code will be included in the next release of Isabelle. It can be downloaded now from the Isabelle development snapshot.[3] The system will be further developed and extended in accordance with users' comments.

The EPSRC project "LEO II: An Effective Higher-Order Theorem Prover" (EP/D070511/1), while not a direct follow-on, is relevant to the current project. The development of a powerful automatic prover for higher-order logic would strengthen our own system, while eliminating the need for (necessarily limited) translations of higher-order problems to first-order logic.

References

[1] Wolfgang Ahrendt, Bernhard Beckert, Reiner Hähnle, Wolfram Menzel, Wolfgang Reif, Gerhard Schellhorn, and Peter H. Schmitt. Integrating automated and interactive theorem proving. In Wolfgang Bibel and Peter H. Schmitt, editors, *Automated Deduction— A Basis for Applications*, volume II. Systems and Implementation Techniques, pages 97–116. Kluwer Academic Publishers, 1998.

[2] Marc Bezem, Dimitri Hendriks, and Hans de Nivelle. Automatic proof construction in type theory using resolution. *Journal of Automated Reasoning*, 29(3-4):253–275, 2002.

[3] Joe Hurd. Integrating Gandalf and HOL. In Yves Bertot, Gilles Dowek, André Hirschowitz, Christine Paulin, and Laurent Théry, editors, *Theorem Proving in Higher Order Logics: TPHOLs '99*, LNCS 1690, pages 311–321. Springer, 1999.

[1] http://www.cl.cam.ac.uk/~lp15/Grants/automation.html

[2] http://www.cl.cam.ac.uk/~lp15/papers/Automation/index.html

[3] http://isabelle.in.tum.de/devel/

[4] Joe Hurd. An LCF-style interface between HOL and first-order logic. In Andrei Voronkov, editor, *Automated Deduction — CADE-18 International Conference*, LNAI 2392, pages 134–138. Springer, 2002.

[5] Joe Hurd. First-order proof tactics in higher-order logic theorem provers. In Myla Archer, Ben Di Vito, and César Muñoz, editors, *Design and Application of Strategies/Tactics in Higher Order Logics*, number NASA/CP-2003-212448 in NASA Technical Reports, pages 56–68, September 2003.

[6] Jia Meng and Lawrence C. Paulson. Translating higher-order problems to first-order clauses. Submitted for journal publication.

[7] Jia Meng and Lawrence C. Paulson. Experiments on supporting interactive proof using resolution. In David Basin and Michaël Rusinowitch, editors, *Automated Reasoning — Second International Joint Conference, IJCAR 2004*, LNAI 3097, pages 372–384. Springer, 2004.

[8] Jia Meng and Lawrence C. Paulson. Translating higher-order problems to first-order clauses. In Geoff Sutcliffe, Renate Schmidt, and Schulz Schulz, editors, *FLoC'06 Workshop on Empirically Successful Computerized Reasoning*, volume 192 of *CEUR Workshop Proceedings*, pages 70–80, 2006.

[9] Jia Meng and Lawrence C. Paulson. Lightweight relevance filtering for machine-generated resolution problems. *Journal of Applied Logic*, in press.

[10] Jia Meng, Claire Quigley, and Lawrence C. Paulson. Automation for interactive proof: First prototype. *Information and Computation*, 204(10):1575–1596, 2006.

[11] Lawrence C. Paulson. A generic tableau prover and its integration with Isabelle. *Journal of Universal Computer Science*, 5(3):73–87, 1999.

[12] Lawrence C. Paulson and Kong Woei Susanto. Source-level proof reconstruction for interactive theorem proving. In Jens Brandt, editor, *Theorem Proving in Higher Order Logics*. Springer, 2007. In press.

[13] Jörg Siekmann, Christoph Benzmüller, Armin Fiedler, Andreas Meier, Immanuel Normann, and Martin Pollet. Proof development with Ωmega: The irrationality of $\sqrt{2}$. In Fairouz Kamareddine, editor, *Thirty Five Years of Automating Mathematics*, pages 271–314. Kluwer Academic Publishers, 2003.

[14] Geoff Sutcliffe, Jürgen Zimmer, and Stephan Schulz. TSTP data-exchange formats for automated theorem proving tools. In W. Zhang and V. Sorge, editors, *Distributed Constraint Problem Solving and Reasoning in Multi-Agent Systems*, number 112 in Frontiers in Artificial Intelligence and Applications, pages 201–215. IOS Press, 2004.