AISB'95 Workshop Report *

Automated Reasoning: Bridging the Gap between Theory and Practice

Andrew Ireland

Department of Artificial Intelligence
University of Edinburgh
80 South Bridge, Edinburgh
Scotland, EH1 1HN
Email: a.ireland@ed.ac.uk

Following the highly successful First Workshop on Automated Reasoning, the aim of this workshop was to provide an informal forum for the automated reasoning community. Researchers across the broad spectrum of automated reasoning have observed a gap between theory and practice in their area of specialization. This workshop aimed to bring together researchers from all areas of automated reasoning to address approaches to bridging this gap. The workshop also aimed to foster links and facilitate cross-fertilization of ideas among researchers from various disciplines; among researchers from academia, industry and government; and between theoreticians and practitioners.

The workshop took place on Tuesday April 4th in Sheffield and was held in conjunction with the tenth AISB Biennial Conference. The workshop attracted over 40 participants. Although AISB is a national conference the workshop was truly international with over a third of the submitted contributions coming from outside the United Kingdom. The technical programme consisted of three panel sessions and an invited speaker. A summary of the workshop together with the wrap-up session is given below.

As Programme Chair I am indebted to my fellow members of the Organizing Committee for their support and guidance: Alan Bundy (Edinburgh), Barry Crabtree (BT Labs, Ipswich), Alan Frisch (Chair, York), Roger Jones (ICL, Wokingham), Antony Galton (Exeter). Thanks also go to Craig MacNish (York) who assisted with publicity.

We would like to thank AISB for sponsoring this workshop and for their organisational help, in particular Robert Gaizauskas. Finally we acknowledge the financial support of the Compulog Net.

Phase Transitions in Constraint Satisfaction (and other) Problems

The panel moderator, Barbara Smith (University of Leeds) introduced the session by outlining what has become known as the phase transition phenomenon in search

*My thanks go to Alan Bundy, Craig MacNish, Alan Frisch, Michael Fisher, Barbara Smith, Toby Walsh, Frieder Stolzenburg, and Paul Curzon for feedback on an earlier draft.

algorithms. That is, the observation that the hardest problems within a randomly generated problem set are located midway between regions of easily soluble and easily insoluble problems and that about half the hardest problems are soluble. While interest in this phenomenon has emerged from the study of constraint satisfaction problems it was stressed that many other classes of problems exhibit a phase transition. The question being addressed by the panel was whether or not this fascinating phenomenon has any practical benefit.

Andrew Davenport (University of Essex) highlighted the fact that recently "exceptionally hard" problems have been found to occur within the easy region of problem spaces. It is still not clear how algorithm dependent this observation is. Andrew presented experimental evidence which suggests that this phenomenon may only be true for complete algorithms. The question arose as to whether or not phase transition results could be used to predict the "best" algorithm for a particular application. Indeed it appears that one can exploit this work in evaluating the trade-offs between, say heuristic and complete algorithms for a given problem set.

By analysing how different algorithms perform as the structure of a problem changes Pat Prossor (University of Strathclyde) pointed out the important role phase transition can play in the design and understanding of algorithms. Moreover he saw a more rigorous approach to comparing and evaluating algorithms as being one of the real practical spin-offs from this work.

Toby Walsh (IRST Trento) widened the discussion by illustrating numerous problems sets outside the domain of constraint satisfaction which give rise to phase transitions. However, he also observed that large scale structure within a problem space can prevent typical phase transition behaviour. This was illustrated by a timetabling problem. While phase transitions are observed widely in random problem sets this raised the question as to how appropriate a predictor it is within "real" problems. The jury is still out on this one.

Can Automated Reasoning Bridge the Formal Methods Gap?

The acceptance of formal methods by industry has been painfully slow. Computer based support for the application of formal methods typically requires a PhD in logic or computer science. Automated reasoning can obviously play a role in overcoming this bottle-neck. The panel, moderated by Alan Bundy (University of Edinburgh), addressed the question of how significant a contribution automated reasoning can make to overcoming this bottle-neck.

Judith Underwood (University of Edinburgh) saw the current bottle-neck as partly due to a conflict between the "ideal" world of the theoretical computer scientist and the "real" world of the programmer. Judith began by outlining the traditional approaches to the formal development of provable correct programs, *i.e.* model based and logic based. Judith highlighted an alternative third approach which makes the link between program and specification direct — an approach which has given rise to the notion of "wide spectrum languages". Judith argued for this rather neglected approach on pragmatic grounds, *i.e.* there is no hope for a "nice" complete proof system for "real" programming languages. Moreover, she stressed that the specification language must be closely related to the implementation language if programmers are going to use it. The real challenge, as Judith saw it, was how existing automated reasoning technology, *e.g.* equational and inductive reasoning, can be integrated within the context of wide spectrum languages.

Through his collaborative work with industry Dieter Hutter (DFKI Saarbrücken) was able to inject some "real world" experience into the proceedings. Dieter outlined VSE, a formal development tool developed by a consortium of universities and industry. The tool which supports the modular development of software following the step-wise refinement paradigm was illustrated via a software development case study. The formal development was carried out in tandem with the conventional developement of the software. On the down side the formal development process took 7 times longer than the conventional development. However, through the use of VSE a bug in the original specification was located. This bug persisted through the conventional development and resulted in the actual project being axed. When asked about the reaction of industry to the development time Dieter observed that it is very important that the formal approach should be restricted to safety and security critical applications. He also highlighted a number of special needs which formal methods imposes upon automated reasoning, *e.g.* structured deduction, heterogeneous calculi and high-level proof strategies which facilitate effective user interaction.

Paul Curzon (University of Cambridge) focused upon the formal verification of hardware. In particular he presented a case study on the formal verification of the Fairisle Asynchronous Transfer Mode communications network switch. Paul observed that the real problems lie not in the difficulty of the individual proofs but rather in the management of the overall proof effort. Paul emphasized the need to develop computer based tools which will effectively take care of the proof management while allowing the user to focus upon the more creative aspects of the verification task. The importance of proof scripts in enabling proof reuse was noted. This adds weight to the argument for using verified generic designs as a means of reducing the cost of verification. This in turn increases the need for support for proof maintenance. Paul noted that this was an area which has been overlooked to a large extent by the community. Other observations Paul touched upon included the need to "design for provability" and the importance of debugging specifications before attempting formal proof.

The need to focus upon specification was reiterated by Craig MacNish (University of York) when he gave a short overview of the PROTEUS Project, which is concerned with the analysis of changing requirements and the problems this poses for requirements capture and representation. Craig's message was that logical representations and automated reasoning are not, on their own, sufficient for the needs of industry. Craig argued for the use of "controlled" natural language as a bridge between design engineers and logical representations.

Role of Automated Deduction in Intellectics *(Invited Talk)*

Wolfgang Bibel delivered a very engaging invited talk which mapped out a dream for a formalism which will unify both classical and non-classical logics. Wolfgang argued that both classical and non-classical formalisms exhibit a basic common structure. The use of the connection method was proposed as a framework for expressing this common structure. He then went on to show how various logics, *e.g.* intuitionistic, default logic *etc*, could be captured by adjustments to the basic structure.

How Should Automated Reasoning Programs Be Evaluated?

Automated Reasoning covers a wide spectrum of research issues within computing. The panel, moderated by Andrew Ireland (University of Edinburgh), considered the issue of evaluating the fruits of our research. Whether one is motivated by programming language design or the modelling of common sense reasoning we must all ultimately address the question of "evaluation".

Drawing upon the logic programming language Gödel, Pat Hill (University of Leeds) focused upon issues of expressivity and efficiency. She showed that increasing the expressivity of one's language need not imply a decrease in efficiency. This was illustrated through the use of restricted quantifications and arrays.

Michael Fisher (Manchester Metropolitan University) considered how evaluation criteria can play a crucial role in the design of automated reasoning systems. In particular, he stressed that objective evaluation is rarely feasible and that automated reasoning systems must be evaluated with respect to their specific target applications. He focused upon the question of how the interaction with theorem provers should be managed. He argued that current interactive theorem provers were not suitable for non-expert users and that the interface should therefore be application dependent, *e.g.* a hardware verification environment should provide an interface which is explicitly tailored to the needs of hardware designers. Alan Bundy (University of Edinburgh) remarked that the design of the underlying theorem prover will greatly constrain the degree to which one can interact. He highlighted three principle concerns, *i.e.* style of proof, proof structuring facilitates and adequate explanation of strategic knowledge.

¿From the stand-point of run-time measures as a means of evaluating theorem provers, Frieder Stolzenburg (University of Koblenz) outlined a comparison among three theorem provers, OTTER, SATCHMO and PROTEIN, on the applications of query answering and interpreters for disjunctive logic programs. In terms of the first application he observed that timings depend upon query structure. More generally timings may depend upon particular problem formalisations and subgoal ordering strategies. The conclusion drawn was that timings alone do not provide a very sophisticated measure.

The session ended with a lively debate on the distinction and interaction between science and technology within automated reasoning.

Wrap-up session

A major objective of this workshop was to attract greater industrial participation. Disappointingly few of the actual participants had industrial experience. This gave rise to a discussion as to how to attract more industrial involvement. One suggestion was to attach future workshops to a more industrial oriented conference. Another suggestion was to actively seek out session topics from industry. One interesting proposal was for an industrial oriented "challenge problem" to be included in the call for participation. Participants would then be invited to detail how they would tackle the problem using automated reasoning technology. Such a proposal, of course, need not be restricted to industrial oriented problems.

It was suggested that panel topics could also be advertised in advance. An alternative suggestion was that the call for panel topics should be made in advance of actual call for participation. A decision on the actual workshop sessions would then be determined by the number of submissions each proposed session topic attracted.

There was strong agreement that the workshop theme and format had worked well. People were keen to see the workshop continue and felt that it played an important role in creating a sense of community among automated reasoning researchers.