1995 AISB Workshop Series

Automated Reasoning: Bridging the Gap between Theory and Practice

Timetable

9.00 - 10.30: **Panel session 1:**
Phase Transitions in Constraint Satisfaction (and other) Problems

10.30 - 11.00: Coffee break

11.00 - 12.45: **Panel session 2:**
Can Automated Reasoning Bridge the Formal Methods Gap?

12:45 - 14:15: Lunch break

14:15 - 15:30: **Invited Talk** – *Wolfgang Bibel:*
The Role of Automated Deduction in Intellectics

15:30 - 16:00: **Panel session 3 (part I):**
How Should Automated Reasoning Programs Be Evaluated?

16:00 - 16:30: Tea break

16:30 - 17:15: **Panel session 3 (part II):**
How Should Automated Reasoning Programs Be Evaluated?

17:15 - 17:45: **Wrap-up session**

Panel Abstracts

Panel 1: *Phase Transitions in Constraint Satisfaction (and other) Problems*

Moderator: Barbara Smith
Panelists: Pat Prosser, Ian Gent, Toby Walsh & Andrew Davenport

Abstract:

Over the last few years it has been recognised that constraint satisfaction problems, as well as many other classes of problem, go through a phase transition, as a parameter is varied, from a region where almost all problems are easy and soluble, to a region where almost all problems are easy and insoluble. In the intervening region, problems are on average much harder, and about half the problems are soluble.

This is a fascinating phenomenon, and worth investigating for that reason. But is it of any practical use to know about the phase transition? If so, what do we need to know about it, and what use can we make of the knowledge? Most of the experiments have been carried out with randomly-generated binary CSPs; do these tell us anything about the behaviour of real problems?

Panel 2: *Can Automated Reasoning Bridge the Formal Methods Gap?*

Moderator: Alan Bundy
Panelists: Judith Underwood, Dieter Hutter & Paul Curzon

Abstract

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 (AR) can obviously play a role in overcoming this bottle-neck. But how significant can we expect this role to be? How should AR techniques be used to ensure the necessary increases in productivity? To what extent can AR techniques be used to reduce the skills bottle-neck which surrounds the use of computer based tools for formal methods?

Panel 3: *How Should Automated Reasoning Programs Be Evaluated?*

Moderator: Alan Frisch
Panelists: Pat Hill & Michael Fisher

Abstract

Automated Reasoning (AR) covers a wide spectrum of research issues within computing. Whether one is motivated by programming language design or the modelling of common sense reasoning we must all ultimately address the question of "evaluation". What, therefore, are the criteria for evaluating our AR programs? How should the relative merits of efficiency versus ease of debugging be balanced? How well do standard problem sets serve our community? What role should case studies play within evaluation? To what extent is the flexibility and adaptability of control strategies important? How can such parameters be measured in a meaningful way? Is "ease of interaction" more or less important than the degree of automation our systems achieve? What are the pit falls to be aware of, *e.g.* to what extent are our strategies sensitive to particular styles of problem definition? Ultimately, does it make sense to aim for some common criteria or principles for evaluating the fruits of AR research?