Dagstuhl-Seminar

on Abstract Interpretation

Organized by:

Patrick Cousot (École Normale Supérieure, Paris)

Radhia Cousot (École Polytechnique, Palaiseau)

Alan Mycroft (Cambridge University)

28 August–1 September 1995

Overview

Abstract interpretation is a mathematical framework for specifying program analyzers which conservatively approximate program behaviours by abstraction of the language semantics. As can be seen from the abstracts of the talks, the seminar succeeded in bringing together researchers from different areas of application of abstract interpretation to focus discussion on various:

- program analysis methods;
- programming paradigms;
- program runtime properties;
- composable abstract domain designs;
- fixpoint computation methods leading to efficient implementations;
- designs of generic program analyzers and compilers using powerful analysis methods, four of them being demonstrated; and
- applications.

The numerous and lively discussions which occurred during the formal sessions, the discussion session on compositionality, and above all the informal meetings confirmed the need for such exchange between the different communities and demonstrated the benefits of this seminar.

The organizers September 1995

Acknowledgements

Thanks are due to Reinhard Wilhelm for inviting us to organize this Dagstuhl seminar on abstract interpretation; to Angelika Müller and Annette Bayer in the Dagstuhl office in Saarbrücken and the staff at Schloß Dagstuhl for ensuring that everything ran perfectly; to the participants for making the seminar lively, fruitful and of very high quality; and to Chris Colby for coordinating the production of this report.

Programme of the Workshop

Monday 28 August 1995

09h00–09h10	**R. Wilhelm**	Organizational remarks
09h10–09h45	**P. Cousot**	Introduction to Abstract Interpretation
09h45–10h30	**W.L. Harrison III**	Abstract Interpretation in a Production C/C++ Compiler
10h30–11h00	*Coffee break*	
11h00–12h00	**B. Ryder**	Flow Sensitive Data Flow Analyses
12h15–14h00	*Lunch*	
14h00–14h45	**F. Martin**	Program Analyzer Generator
14h45–15h15	**P. Cousot**	Compositional Design of Galois Connections
15h15–16h00	**C. Fecht**	A Tool for Generating Efficient Prolog Analyzers
16h00–16h30	*Coffee break*	
16h30–17h30	**G. Filé**	Cut, Complement and Paste with Abstract Domains
17h30–18h00	**P. Cousot**	Fixpoint Approximation with Widening/Narrowing
18h00–19h00	*Dinner*	
19h00–20h00	*U. Loebens*	*Opening of U. Loebens Art Exhibition*

Tuesday 29 August 1995

09h00–09h45	**G. Levi**	Abstract Debugging
09h45–10h45	**M. Codish**	Bottom-Up Abstract Interpretation of Logic Programs—from Theory to Practice
10h45–11h45	*Coffee break*	
11h45–12h00	**M. Hanus**	Abstract Interpretation of Functional Logic Languages
12h00–12h15	**M. Sagiv**	A Few Words on Interprocedural Analysis
12h15–14h00	*Lunch*	
14h00–14h30	**C. Hankin**	Analysis of Value-Passing Process Calculi
14h30–15h30	**F. Nielson**	Inference Based Analysis
15h30–16h15	**J. Palsberg**	A Type System Equivalent to Flow Analysis
16h15–16h45	*Coffee break*	
16h45–17h45	**M. Bruynooghe**	A Framework for Abstract Interpretation of Logic Programs Based on Preinterpretations
18h00–19h00	*Dinner*	
20h00–21h00	**F. Martin**	Demo: Program Analyzer Generator
	C. Fecht	Demo: Prolog Analyzer Generator

Wednesday 30 August 1995

09h00–09h45	**N. Jones**	Optimizing Run-Time Data and Operations
09h45–10h30	**F. Bourdoncle**	Implicit Higher-Order Polymorphism with Primitive Typing
10h30–11h00	*Coffee break*	
11h00–11h45	**D. Boulanger**	Decoding Models of Logic Programs
11h45–12h15	**P. Granger**	Local Iterations
12h15–14h30	*Lunch*	
14h30–18h00		*Visit to Trier/Hiking/Cycling*
18h00–19h00	*Dinner*	
20h00–21h00	**F. Bourdoncle**	Demo: SYNTOX: A Pascal Abstract Interpreter

Thursday 31 August 1995

09h00–09h45	**H. Søndergaard**	Dependency Analysis for Logic Programs
09h45–10h30	**R. Shyamasundar**	Termination Analysis of Logic Programs
10h30–11h00	*Coffee break*	
11h00–11h45	**M. Rosendahl**	On Fixpoint Iteration in Abstract Interpretation
11h45–12h15	**T. Jensen**	Flow Analysis in the Geometry of Interaction
12h15–14h30	*Lunch*	
14h30–15h15	**M. Sagiv**	Graph-based Pointer Analysis via Abstract Interpretation
15h15–16h00	**A. Deutsch**	Semantic Models of Dynamic Storage and their Abstractions
16h00–16h30	*Coffee break*	
16h30–17h15	**K. Marriott**	Practical Experience in Using Abstract Interpretation in a Highly Optimizing Compiler for CLP(R)
17h15–18h00	**H. Seidl**	Set Constraints to Stop Deforestation
18h00–19h00	*Dinner*	
19h00–19h30	**W.L. Harrison III**	Demo: C/C++ Abstract Interpretor
19h30–21h00	All participants	Discussion

Friday 1 September 1995

09h00–09h30	**U. Assmann**	Abstract Interpretation and Data Flow Analysis with Graph Rewrite Systems
09h30–10h00	**A. Poetzsch-Heffter**	Interprocedural Data Flow Analysis Based on Temporal Specifications
10h00–10h30	**O. Rüthing**	The Complexity of Exhaustive Motion-Elimination Frameworks
10h30–11h00	*Coffee break*	
11h00–11h30	**J. Knoop**	Analyzing and Optimizing Parallel Programs
11h30–12h00	**C. Colby**	Trace-Based program Analysis
12h15–14h30	*Lunch*	

Abstracts of the talks