University Paris Sud (Orsay) site and Grenoble and France Telecom R& D subsites

C. Paulin

1/09/2004-31/08/2005

1 Major scientific results

1.1 Correctness of Computer Systems

Proving C or Java programs Our main activity is related to program verification. We mainly focus on the verification of behavioral specifications for programming languages such as C, Java and ML. We develop a tool "Why" which is a verification conditions generator: from an annotated program written in a small imperative language with Hoare logic-like specification, it generates conditions expressing the correctness and termination of the program. These verification conditions can be generated for several existing provers, including interactive proof assistants (Coq, PVS, HOL Light, Mizar) and automatic provers (Simplify, haRVey, CVC Lite).

On top of this tool, we built a system called Krakatoa [3] which verifies Java source code annotated with the Java Modeling Language (JML). The main challenge was the design of a suitable model for the Java memory heap in order to tackle programs with possible aliases [6].

J.-C. Filliâtre and C. Marché designed a similar tool called Caduceus [3] for dealing with C programs. This tool was used by Th. Hubert and C. Marché [4] for proving a subtle algorithm due to Schorr & Waite for graphs traversal. J. Andronick [1] experimented on using this tool for formal verification of security properties of smart card embedded source code.

Timed automata Orsay and France Telecom R& D collaborated on the definition of a model of timed automata in Coq. It is integrated in the CALIFE platform, a general tool for specification and automatic or interactive verification of protocols. We are currently studying the quantitative analysis of behavior of protocols built on random choices.

Dependent types For his master work supervised by C. Paulin, M. Sozeau [4] designed a language with a subset type (in the spirit of the PVS language) which is convenient for programming with (a restricted class of) dependent types. He proposed a translation of a term in this language to a Coq term containing existential variables corresponding to type-checking conditions.

Case studies We developed several case studies in Coq related to correctness of computer systems.
J.-F. Monin [7] from Grenoble subsite proved that the functional sprintf fonction of Danvy and the usual version of sprintf (with a dependent typing) are intensionally equal. Th. Hubert [3] developed libraries for certifying termination proofs using dependent pairs criteria in Coq.

1.2 Formal Mathematics and Mathematics Education

J. Duprat together with L. Vuillon from the Chambery subsite is working on formalizing discrete geometry by inductive objects.

1.3 Proof Technology

For his master work supervised by J-C. Filliâtre, N. Ayache [1] designed an interactive tactic for calling first-order automatic provers from the Coq proof assistant. The main difficulty was to derive an apropriate first-order theory from an higher-order environment.

1.4 Foundational Research

Automatic deduction Integrating automatic deduction into type theory is a long term research. P. Corbineau [2, 2] made a significant contribution extending results in first-order intuitionistic logic with equality to the case of predicate defined by constructors. S. Conchon [1, 5] is studying decision procedures adapted to automatic resolution of proof obligations generated by checking correctess of programs.

Extensionality Mathematical proofs make an implicit use of extensionality which identify two objects which are provably equal. N. Oury [8] studied this rule and proposed a translation of a derivation in an extensional sytem into an intensional proof in a system like Coq.

2 Cooperation with industry

We are collaborating with Dassault Aviation in the area of proofs of C programs. We also have a collaboration with the Axalto company (a smartcards manufacturer) on proofs of Java and C programs, Javacard applets and operating systems. Th. Hubert (Dassault), J. Andronick and N. Rousset (Axalto) are studying for their PhD part-time in the industry and part-time in our laboratory.

There is a collaboration between Orsay, Grenoble and the industrial subsite France Télécom R& D in the AVERROES national project (analysis and verification for the reliability of embedded systems) (`http://www-verimag.imag.fr/AVERROES`).

We also have a collaboration with César Muñoz at NIA, Hampton, USA on proof of Java programs for avionics.

We participate to the new competitiveness cluster System@tic (`http://www.systematic-paris-region.org`). In this cluster, the main industrial and academic research centers in the Ile-de-France Region are collaborating in the area of complex systems.

3 List of visitors from other sites

D. Walukiewiecz and J. Chrząszcz from Warsaw University visited our site in June 2005. They gave a seminar on "Consistency and Completeness of Rewriting in the Calculus of Constructions". Patryk Czarnik a Phd student from Warsaw University visited the Orsay group 4 days in april 2005 and gave a seminar on his experiment using the Krakatoa tool for proving Java programs.

Sites interactions We have many exchanges (including a common seminar) with the LogiCal project which is part of the INRIA site.

Visit to other sites Jean-Christophe Filliâtre visited Queen Mary site in april 2005 and gave a seminar on "Verifying C and Java programs".

3.1 URL

`http://www.lri.fr/demons/introduction.en.html`

4 List of publications

Refereed journal papers

[1] Sylvain Conchon and Sava Krstić. Strategies for combining decision procedures. *Theoretical Computer Science*, 2004. Special Issue of TCS dedicated to a refereed selection of papers presented at TACAS'03.

[2] Evelyne Contejean, Claude Marché, Ana Paula Tomás, and Xavier Urbain. Mechanically proving termination using polynomial interpretations. *Journal of Automated Reasoning*, 2005.

[3] Claude Marché, Christine Paulin-Mohring, and Xavier Urbain. The Krakatoa tool for certification of Java/JavaCard programs annotated in JML. *Journal of Logic and Algebraic Programming*, 58(1–2):89–106, 2004. `http://krakatoa.lri.fr`.

[4] Claude Marché and Xavier Urbain. Modular and incremental proofs of AC-termination. *Journal of Symbolic Computation*, 38:873–897, 2004.

[5] Sava Krstić and Sylvain Conchon. Canonization for disjoint unions of theories. *Information and Computation*, 2004. Special Issue of Information and Computation dedicated to a refereed selection of papers presented at CADE-19.

Refereed conferences papers

[1] June Andronick, Boutheina Chetali, and Christine Paulin-Mohring. Formal verification of security properties of smart card embedded source code. In John Fitzgerald, Ian J. Hayes, and Andrzej Tarlecki, editors, *International Symposium of Formal Methods Europe (FM'05)*, volume 3582 of *Lecture Notes in Computer Science*, Newcastle,UK, July 2005. Springer-Verlag.

[2] Evelyne Contejean and Pierre Corbineau. Reflecting proofs in first-order logic with equality. In *20th International Conference on Automated Deduction (CADE-20)*, Lecture Notes in Computer Science, Tallinn, Estonia, July 2005. Springer-Verlag.

[3] Jean-Christophe Filliâtre and Claude Marché. Multi-prover verification of C programs. In Jim Davies, Wolfram Schulte, and Mike Barnett, editors, *Sixth International Conference on Formal Engineering Methods*, volume 3308 of *Lecture Notes in Computer Science*, pages 15–29, Seattle, WA, USA, November 2004. Springer-Verlag.

[4] Thierry Hubert and Claude Marché. A case study of C source code verification: the Schorr-Waite algorithm. In *3rd IEEE International Conference on Software Engineering and Formal Methods (SEFM'05)*, Koblenz, Germany, September 2005.

[5] Bart Jacobs, Claude Marché, and Nicole Rauch. Formal verification of a commercial smart card applet with multiple tools. In *Algebraic Methodology and Software Technology*, volume 3116 of *Lecture Notes in Computer Science*, Stirling, UK, July 2004. Springer-Verlag.

[6] Claude Marché and Christine Paulin-Mohring. Reasoning about Java programs with aliasing and frame conditions. In Hurd and Melham [1].

[7] Jean-François Monin. Proof pearl: From concrete to functional unparsing. In K. Slind, A. Bunker, and G. Gopalakrishnan, editors, *International Conference on Theorem Proving in Higher Order Logics (TPHOLs 2004)*, volume 3223 of *Lecture Notes in Computer Science*, pages 217–224. Springer-Verlag, Park City, Utah, USA, September 2004.

[8] Nicolas Oury. Extensionality in the Calculus of Constructions. In Hurd and Melham [1].

PhD and Master thesis

[1] Nicolas Ayache. Coopération d'outils de preuve interactifs et automatiques. Master's thesis, Université Paris 7, 2005.

[2] Pierre Corbineau. *Démonstration Automatique en Théorie des Types*. Thèse de doctorat, Université Paris-Sud, September 2005.

[3] Thierry Hubert. Certification des preuves de terminaison en Coq. Rapport de DEA, Université Paris 7, September 2004. In French.

[4] Matthieu Sozeau. Coercion par prédicats en Coq. Master's thesis, Université Paris 7, 2005.

Manual

[1] The Coq Development Team. *The Coq Proof Assistant Reference Manual – Version V8.0*, April 2004. `http://coq.inria.fr`.

Cross-references

[1] J. Hurd and T. Melham, editors. *Theorem Proving in Higher Order Logics: 18th International Conference, TPHOLs 2005*, Lecture Notes in Computer Science. Springer-Verlag, August 2005.