Activity Report 2014

Project-Team MARELLE

Mathematical Reasoning and Software

RESEARCH CENTER
Sophia Antipolis - Méditerranée

THEME
Proofs and Verification

Table of contents

Project-Team MARELLE

Keywords: Interactive Theorem Proving, Formal Methods, Security, Cryptography

Creation of the Project-Team: 2006 November 01.

1. Members

Research Scientists

Yves Bertot [Team leader, Inria, Senior Researcher, HdR]
Cyril Cohen [Inria, Researcher, from October 2014]
José Grimm [Inria, Researcher]
Benjamin Grégoire [Inria, Researcher]
Laurence Rideau [Inria, Researcher]
Enrico Tassi [Inria, Researcher, from September 2014]
Laurent Théry [Inria, Researcher]

Visiting Scientists

Amy Felty [Professor at University of Ottawa, Canada, on sabbatical leave, until Aug 2014]
Douglas Howe [Professor at Carleton University, Canada, until Jul 2014]

Administrative Assistant

Nathalie Bellesso [Inria]

Others

Sophie Bernard [ENS Lyon, Master-2 student, from Mar 2014 until Aug 2014]
Guillaume Cano [Doctoral student, Inria, until Mar 2014]
Loic Pottier [Min. de l'Education Nationale, Researcher, HdR]

2. Overall Objectives

2.1. Overall Objectives

We want to concentrate on the development of mathematical libraries for theorem proving tools. This objective contributes to two main areas of application: tools for mathematicians and correctness verification tools for software dealing with numerical computation.

In the short term, we aim for mathematical libraries that concern polynomials, algebra, group theory, floating point numbers, real numbers, big integers, probabilities and geometrical objects. In the long run, we think that this will involve any function that may be of use in embedded software for automatics or robotics (in what is called hybrid systems, systems that contain both software and physical components) and in cryptographical systems. We want to integrate these libraries in theorem proving tools because we believe they will become important tools for mathematical practice and for engineers who need to prove the correctness of their algorithms and software.

We believe that theorem proving tools are good tools to produce highly dependable software, because they provide a framework where algorithms and specifications can be studied uniformly and often provide means to mechanically derive programs that are correct by construction.

We also study the extensibility of interactive theorem proving tools based on decision procedures that free designers from the burden of verifying some of the required properties. We often rely on "satisfiability modulo theory" procedures, which can be connected to theorem proving tools in a way that preserves the trustability of the final results.

3. Research Program

3.1. Type theory and formalization of mathematics

The calculus of inductive constructions is a branch of type theory that serves as a foundation for theorem proving tools, especially the Coq proof assistant. It is powerful enough to formalize complex mathematics, based on algebraic structures and operations. This is especially important as we want to produce proofs of logical properties for these algebraic structures, a goal that is only marginally addressed in most scientific computation systems.

The calculus of inductive constructions also makes it possible to write algorithms as recursive functional programs which manipulate tree-like data structures. A third important characteristic of this calculus is that it is also a language for manipulating proofs. All this makes this calculus a tool of choice for our investigations. However, this language is still being improved and part of our work concerns these improvements.

3.2. Verification of scientific algorithms

To produce certified algorithms, we use the following approach: instead of attempting to prove properties of an existing program written in a conventional programming language such as C or Java, we produce new programs in the calculus of constructions whose correctness is an immediate consequence of their construction. This has several advantages. First, we work at a high level of abstraction, independently of the target implementation language. Secondly, we concentrate on specific characteristics of the algorithm, and abstract away from the rest (for instance, we abstract away from memory management or data implementation strategies). Therefore, we are able to address more high-level mathematics and to express more general properties without being overwhelmed by implementation details.

However, this approach also presents a few drawbacks. For instance, the calculus of constructions usually imposes that recursive programs should explicitly terminate for all inputs. For some algorithms, we need to use advanced concepts (for instance, well-founded relations) to make the property of termination explicit, and proofs of correctness become especially difficult in this setting.

3.3. Programming language semantics

To bridge the gap between our high-level descriptions of algorithms and conventional programming languages, we investigate the algorithms that are present in programming language implementations, for instance algorithms that are used in a compiler or a static analysis tool. For these algorithms, we generally base our work on the semantic description of a language. The properties that we attempt to prove for an algorithm are, for example, that an optimization respects the meaning of programs or that the programs produced are free of some unwanted behavior. In practice, we rely on this study of programming language semantics to propose extensions to theorem proving tools or to participate in the verification that compilers for conventional programming languages are exempt from bugs.

4. Application Domains

4.1. Reliability of embedded software

Software embedded in physical devices performs computations where the inputs are provided by measures and the outputs are transformed into actions performed by actuators. to improve the quality of these devices, we expect that all the computations performed in this kind of software will need to be made more and more reliable. We claim that formal methods can serve this purpose and we develop the libraries and techniques to support this claim. This implies that we take a serious look at how mathematics can be included in formal methods, especially concerning geometry and calculus.

4.2. Security and Cryptography

The modern economy relies on the possibility for every actor to trust the communications they perform with their colleagues, customers, or providers. We claim that this trust can only be built by a careful scrutiny of the claims made by all public protocols and software that are reproduced in all portable devices, computers, and internet infrastructure systems. We advocate the use of formal methods in these domains and we provide easy-to-use tools for cryptographers so that the formal verification of cryptographic algorithms can become routine and amenable to public scrutiny.

4.3. Mathematics and Education

As librairies for theorem provers evolve, they tend to cover an ever increasing proportion of the mathematical background expected from engineers and scientists of all domains. Because the content of a formally verified library is extremely precise and explicit, we claim that this will provide a new kind of material for teaching mathematics, especially useful in remote education.

5. New Software and Platforms

5.1. Coq

Participants: Enrico Tassi, Benjamin Grégoire.

Coq is developed mainly in the project-team $\pi.r^2$ with contributions from many other individuals. Enrico Tassi and Benjamin Grégoire are regular contributors. In particular for 2014, Benjamin Grégoire provided advice on connecting virtual machine execution with other aspects of the Coq system and Enrico Tassi worked on a new interactive mode that supports a *document* view of the proof script, with faster user experience. Enrico Tassi also worked on improvements for the use of Coq on Windows.

5.2. Easycrypt

Participants: Gilles Barthe [IMDEA Software Institute], François Dupressoir [IMDEA Software Institute], Benjamin Grégoire [correspondant], César Kunz [IMDEA Software Institute], Benedikt Schmid [IMDEA Software Institute], Pierre-Yves Strub [IMDEA Software Institute].

EasyCrypt is a toolset for reasoning about relational properties of probabilistic computations with adversarial code. Its main application is the construction and verification of game-based cryptographic proofs. EasyCrypt can also be used for reasoning about differential privacy.

5.3. zoocrypt

Participants: Gilles Barthe [IMDEA Software Institute], François Dupressoir [IMDEA Software Institute], Benjamin Grégoire [correspondant], César Kunz [IMDEA Software Institute], Benedikt Schmid [IMDEA Software Institute], Pierre-Yves Strub [IMDEA Software Institute].

ZooCrypt (see http://www.easycrypt.info/zoocrypt/) is an automated tool for analyzing the security of padding-based public-key encryption schemes (i.e. schemes built from trapdoor permutations and hash functions). This years we extended the tool to be able to deal with schemes based on cyclic groups and bilinear maps.

5.4. CoqApprox

Participants: Nicolas Brisebarre [CNRS], Mioara Joldes, Érik Martin-Dorel, Micaela Mayero [Iut de Villetaneuse], Jean-Michel Muller, Ioana Paşca [Iut de Nimes], Laurence Rideau, Laurent Théry [correspondant].

We develop a formalization of rigorous polynomial approximation using Taylor models inside the Coq proof assistant, with a special focus on genericity and efficiency for the computations. In 2014, this library has been included in CoqInterval, distributed by the Toccata research team.

5.5. Ssreflect and Mathematical Components

Participants: Yves Bertot, Cyril Cohen, Laurence Rideau, Enrico Tassi [correspondant], Laurent Théry.

Most of the formal proofs developed in our team are integrated in the Ssreflect extension of the Coq system and the Mathematical Components library. Work this year has concentrated on providing new versions of ssreflect that are compatible with the evolutions of Coq (to prepare for the upcoming release) and integrating our results in the description of real numbers. We also laid the foundations for a book explaining the structure and principles at work in the Math-components library.

6. New Results

6.1. Highlights of the Year

In June 2014, Yves Bertot received the ACM Software System award, as one of the main contributors to the Coq System, along with Gérard Huet, Thierry Coquand, Christine Paulin-Mohring, Bruno Barras, Jean-Christophe Filliâtre, Hugo Herbelin, Chet. Murthy, and Pierre Castéran.

6.2. Proof and computation

Participants: Laurent Théry [correspondant], Benjamin Grégoire.

We have been continuing our effort to improve the computing power of Coq. This has led to two "computational proof":

The Erdös conjecture for n = 2 was proved this year using a SAT solver. We succeeded to formally prove this instance in Coq independently checking the 3Gb trace of the SAT solver.

The weak Goldbach conjecture was proved last year by Harald Helfgott. This proof requires a computation that the conjecture holds for numbers less than 10^{28}. This is done in two stages. The first one is to verify Goldbach conjecture for numbers less than 10^{18}. The second one is to verify the weak Goldbach conjecture for numbers less than 10^{28} using a ladder with intervals 10^{18}. The second stage has been completely verified in Coq. We are currently working on improving the computation power of Coq to make it possible to perform the first stage in reasonable time.

6.3. Formal verification of automated proof algorithms

Participant: Laurent Théry [correspondant].

We have been interested in proving that the classic 2-Sat problem can be solved in linear time. This leads to proving two classic algorithms:

1. A version of Kosaraju's algorithm that computes the strongly connected components of a directed graph [21],
2. A more direct algorithm that solves the 2-Sat problem that is using unit propagation, proposed by Alvaro del Val [20].

6.4. Formal study of cryptography

Participants: Gilles Barthe [IMDEA], Sonia Belaid [THALES and ENS], François Dupressoir [IMDEA], Pierre-Alain Fouque [Université de Rennes 1 and Institut universitaire de France], Cédric Fournet [Microsoft Research], Benjamin Grégoire [correspondant], Benedikt Schmidt [IMDEA], Pierre-Yves Strub [IMDEA], Nikhil Swamy [Microsoft Research], Mehdi Tibouchi [NTT Secure Platform Laboratories], Santiago Zanella-Béguelin [Microsoft Research], Jean-Christophe Zapalowicz [Inria].

The goal of this work is to provide a friendly tool easily usable by cryptographers without knowledge of formal proof assistants. The idea is to use the techniques formally proved in Certycrypt and to call SMT-provers. We provide two differents tools EasyCrypt and ZooCrypt.

This year, we worked on the following topics:

- Relational program logics, as used in EasyCrypt, have been used for mechanizing formal proofs of various cryptographic constructions. In [15], we present rF*, a relational extension of F*, a general-purpose higher-order stateful programming language with a verification system based on refinement types. The distinguishing feature of rF* is a relational Hoare logic for a higher-order, stateful, probabilistic language.
- Fault Attacks are attacks in which an adversary with physical access to a cryptographic device, say a smartcard, tampers with the execution of an algorithm to retrieve secret material. In [13] we propose a new approach for finding fault attacks based on fault conditions. Using the method, we discover multiple fault attacks on RSA and ECDSA. Several of the attacks found by our tool are new. In [14], we propose a new counter measure to make RSA-PSS provably secure against non-random faults. We also prove the result using EasyCrypt.
- Many algorithms, particularly in cryptography, admit very efficient batch versions that compute simultaneously the output of the algoritms on a set of inputs. AutoBatch is a tool that computes highly optimized batch verification algorithms for pairing based signature schemes. In [12], we use EasyCrypt to formalise the methods used by AutoBatch and to automatically certify the result of the transformation performed by AutoBatch.
- We study the problem of automatically verifying higher-order masking countermeasures which is used to protect implementations where the attacker can observe intermediate computations (like in a smartcard). We propose an efficient method to check the correctness and the security of masked implementation. This work has been submitted to EuroCrypt 2015. We start the ANR BRUTUS on this subject.

6.5. Formalization of Bourbaki's sets and ordinals

Participant: José Grimm.

In previous years we developed a formal library describing the parts of the Bourbaki books on set theory, cardinals and ordinals. We completed it by adding the definition of real numbers using Dedekind cuts. The important properties we showed that $\mathbf{R}$ is an ordered Archimedean field, that every non-empty bounded subset has a least upper bound, that every Cauchy sequence has a limit, and that the intermediate value theorem holds.

It follows that every positive real number has positive square root. We give a pair of adjacent sequences that converges to this square root. For instance $\sqrt{2}$ is irrational, and we get a pair of rational adjacent sequences that converges to it. This produces an explicit order isomorphism $\mathbf{Q}^* \to \mathbf{Q}$. The number of such isomorphisms is equal to the power of the continuum (the cardinal of $\mathbf{R}$) [18].

6.6. Stern-Brocot and Fibonacci sequences

Participant: José Grimm.

We constructed an explicit bijection $\mathbf{N} \to \mathbf{Q}$, first in the framework of the Bourbaki project (see above), then in Ssreflect. Every positive rational number x can uniquely be written as a quotient s_n/s_{n+1}. This result was established by Dijkstra who stated it in an obfuscated way. It was shown years before by Stern. It is possible to compute s_n/s_{n+1} without computing numerator and denominator separately, by considering the sequences of bits of n from left to right or from right to left. Truncating the binary expansion of n yields a sequence of approximations to s_n/s_{n+1} (this was studied by Brocot, and the so-called Stern-Brocot tree is an alternative representation of rational numbers). We implemented the work of Dijkstra and Stern in Coq [17].

We also studied how a number can be represented by a sequence of other numbers (for instance as a sum of distinct Fibonacci numbers, with or without constraints). The number of ways of writing n as a sum of powers of two, each power of two being used at most twice, is s_{n+1}. These results are presented in [17].

6.7. Formal proof that e and π are transcendental

Participants: Sophie Bernard, Laurence Rideau.

We constructed formal proofs that π is irrational, e is transcendental, and π is transcendental. These proofs share a common initial pattern, where rationality or algebraicity of the mathematical constants are shown to imply the existence of a sequence of positive integers that must decrease indefinitely.

This proof development is an opportunity to study the interplay between several existing libraries about algebraic structures and analysis: the ssreflect library for algebra and the Coquelicot library for calculus. Moreover, the proof that π is transcendental was an occasion to test the newly developed module on symmetric polynomials by P.-Y. Strub at IMDEA.

6.8. Fast computation of π

Participant: Yves Bertot.

In the previous year, we studied a proof that π could be approximated with a fast converging sequence based on arithmetic geometric means. This year we described a proof that rounding errors during this computation could be guaranteed as small as needed, based on a study of derivatives. This approach provides a fruitful alternative to interval-based approaches. The result was published in [16].

We also completed a journal paper on various ways to observe and compute the number π [7].

6.9. Decision procedures for polynomials

Participant: Yves Bertot.

Following up on the work in previous years around Bernstein Polynomials, we implemented a decision procedure for guaranteeing the sign of a polynomial function inside an interval, using Bernstein polynomials and dichotomy. In the long run, we hope to explore two approaches, one based on the off-line computation of certificates for sub-intervals (these certificates are easy to verify), and one based on implementing computational reflection. This approach should also generalize quite easily to multi-variate polynomials.

7. Bilateral Contracts and Grants with Industry

7.1. Bilateral Contracts with Industry

7.1.1. Collaboration within the Inria/Microsoft Research Joint Centre

We participate in the collaboration *Mathematical Components 2* with Microsoft Research. Currently, the main thrust lies around the exploitation of results in the Mathematical Components library, which was our main point of focus until the completion of the proof of the Feit-Thompson theorem.

8. Partnerships and Cooperations

8.1. National Initiatives

8.1.1. ANR

In 2014, we participated to two successful applications for funding from the French national agency for research (ANR).

- BRUTUS "Chiffrements authentifiés et résistants aux attaques par canaux auxiliaires", started on October 1st, 2014, for 60 months, with a grant of 41 kEuros for Marelle. Other partners are Université de Rennes 1, CNRS, secrétariat Général de la défense et de la sécurité nationale, and Université des Sciences et Technologies de Lille 1. The corresponding researcher for this contract is Benjamin Grégoire.
- FastRelax, "Fast and Reliable Approximations", started on October 1st, 2014, for 60 months, with a grant of 75 kEuros for Marelle. Other partners are Inria Grenoble (ARIC project-team), LAAS-CNRS (Toulouse), Inria Saclay (Toccata and Specfun project-teams), and LIP6-CNRS (Paris). The corresponding researcher for this contract is Laurence Rideau.

8.2. International Initiatives

8.2.1. Inria International Partners

8.2.1.1. Informal International Partners

Our main partner for work on Ssreflect is Georges Gonthier, senior researcher at Microsoft Research, Cambridge.

Our team has important discussions with the team of Thierry Coquand at *Chalmers University and University of Göteborg*. This was illustrated in the past by the European project FORMATH, in the context of which we collaborated around the formalization of various aspects of Algebra (linear algebra and algebraic topology). This effort was continued in the context of the international effort around *homotopy type theory*, where Cyril Cohen is deeply involved (in particular in the implementation of a model for cubical sets). In the future, we may hope to play a continuing role in *homotopy theory* and establish more contacts with other sites involved in this topic.

We participate in the international development of the Coq community and maintain frequent contacts with the most active users around the world. In practice, this implies many contacts with several universities in the United States of America: Princeton University, University of Pennsylvania, the Massachusetts Institute of Technology, Harvard University, and Yale University.

We have intensive collaborations with IMDEA, Madrid. In particular, the software systems EasyCrypt and ZooCrypt are developed in collaboration with this institution, and several of our publications are co-authored between Inria and IMDEA.

8.3. International Research Visitors

8.3.1. Visits of International Scientists

8.3.1.1. Sabbatical programme

Amy Felty, professor at University of Ottawa, was a member of our team until September 30th, on sabbatical leave from her university, and with no extra financial support from Inria.

Dough Howe, professor at Carleton University, was a member of our team until August 31st, on sabbatical leave from his university, and with no extra financial support from Inria.

9. Dissemination

9.1. Promoting Scientific Activities

9.1.1. Scientific events selection

9.1.1.1. Member of the conference program committee

Benjamin Grégoire was member of the program committee for the conference TCS (Theoretical Computer Science), which was held on September 1-3 in Roma, Italy.

Yves Bertot was member of the programm committee for the conference ITP (Interactive Theorem Proving), which was held July 14-17 in Vienna, Austria and for the workshop UITP (User Interfaces for Theorem Provers), which was on held on July 17th in Vienna, Austria.

Laurent Théry was member of the programm committee for the conferences ITP (Interactive Theorem Proving) and UITP (User Interfaces for Theorem Provers) in Vienna (see above) and for the workshop ACL2 (International Workshop on the ACL2 Theorem Prover and its Applications), which was held July 12-13 in Vienna, Austria.

9.1.1.2. Reviewer

Members of the team reviewed papers for the conferences CPP (Certified Programs and Proofs), CCS (ACM Conference on Computer and Communication Security), CSF (Computer Security Foundations), Types (Types for Proofs and Programs),.

9.1.2. Journal

9.1.2.1. Member of the editorial board

Laurent Théry was a special issue editor for Mathematics in Computer Science, "Special Focus on Formal Proofs for Mathematics and Computer Science".

9.1.2.2. Reviewer

Members of the team reviewed papers for the journals JFP (Journal of Functional Programming), TOPLAS (Transactions on Programming Languages and Systems), MSCS (Mathematical Structures in Computer Science), JFR (Journal of Formalized Reasoning), TCS (Journal of Theoretical Computer Science), AMAI (Annals of Mathematics and Artificial Intelligence), and SCP (Science of Computer Programming).

9.1.2.3. Evaluation of funded grants

Laurence Rideau evaluated a grant proposal for the Digiteo laboratory (based in Saclay).

Yves Bertot evaluated a grant proposal for the European Research Concil (consolidator grants).

Laurent Théry evaluated a grant proposal for the French Research Agency (ANR).

9.2. Participation in scientific events

9.2.1. Keynotes, tutorials, and invited talks

Yves Bertot gave an invited talk at the conference CICM (Conference on Intelligent Computer Mathematics) in Coimbra, Portugal, in July.

Laurent Théry, Laurence Rideau, and Yves Bertot gave invited talks at the workshop "Mathematical Structures of Computation", special week on "Formal Proof, Symbolic Computation and Computer Arithmetic", in Lyon, France, in February.

Yves Bertot gave talks at the University of Edinburgh in June, at the NASA Ames research center in California, in June, at SRI in California, in June, at the University of Aveiro in July, and at the University of Tokyo in September. He also attended the ACM Award ceremony in San Francisco, California, in June.

9.3. Teaching - Supervision - Juries

9.3.1. Teaching

Licence : Laurence Rideau, "programming and algorithms", 50 hours, Lycée Masséna, Nice, France.

Master : Yves Bertot, "software verification and computer proof", 21 hours, Master, Université de Nice, France.

Master : Laurent Théry, "introduction to Coq", 3 hours, Ecole des Mines de Paris, France.

Doctorate : Yves Bertot, Formal proofs in coq, 10 hours, NII Shonan, Japan

Doctorate : Benjamin Grégoire, Proofs in Easycrypt, 9 hours, Inria, France

9.3.2. Supervision

PhD : Guillaume Cano, "Intéraction entre algèbre linéaire et analyse en mathématiques formelles", Université de Nice, 4 avril 2014, supervised by Yves Bertot [19].

9.3.3. Juries

Yves Bertot was examiner with written report duty for the thesis of Pierre Boutillier (Université de Paris-Diderot, France, February 18th), external examiner for the thesis of Phil Scott (university of Edinburgh, Scotland, June 4th), and examiner with written report duty for the Habilitation of Sylvie Boldo (Université de Paris-Sud, France, October 6th).

Laurent Théry was examiner for the thesis of Anders Mörtberg (Chalmers University, Sweden, December 12th), This thesis was partially supervised by Cyril Cohen.

9.3.4. Community service

- José Grimm is a member of the comité de centre, the committee where representatives of personnel and management discuss questions of daily life at the level of the Sophia-Antipolis Méditerranée center, he also participates in a commision on continued training and a commission on hygiene, safety, and working conditions. This activity involves around 12 meetings per year.
- Benjamin Grégoire is a member of the *comittee of users of information technology* (CUMI). This activity involves monthly meetings where problems in using the IT infrastructure are debated between researchers and engineers.
- Laurent Théry is a member of the *comité de développement technologique* (in English, technological development committee), the committee that oversees the allocation of software engineers on experimental software and platform development.
- Laurent Théry is a member of the committee that provides advice to the center director concerning the attribution of grants for doctoral students, post-doctoral researchers, and invited professors.
- Yves Bertot is the chairman of the *Coq Steering committee*. Benjamin Grégoire was a member of this committee until October and has been replaced by *Enrico Tassi*. These task imply continuous monitoring of the evolution of Coq, the relations with users, and participating in strategic decisions.
- Yves Bertot was deputy scientific director for the Sophia Antipolis méditerranée research center until August. This task implies meetings approximately every fortnight with the center director, the scientific director, and the director of admnistrative services for the center, together with frequent meetings with researchers from any domain in the center and monthly meetings at the national level as part of the evaluation committee. Yves Bertot still participate to the evaluation committee at national level.

9.4. Popularization

Laurent Théry presented his researcher work to three different classrooms during the event "semaine des maths", in March.

10. Bibliography

Major publications by the team in recent years

[1] G. BARTHE, B. GRÉGOIRE, S. HERAUD, S. Z. BÉGUELIN. *Computer-Aided Security Proofs for the Working Cryptographer*, in "Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings", Lecture Notes in Computer Science, Springer, 2011, vol. 6841, pp. 71-90, Best Paper Award

[2] Y. BERTOT, P. CASTÉRAN. *Interactive Theorem Proving and Program Development, Coq'Art:the Calculus of Inductive Constructions*, Springer-Verlag, 2004

[3] Y. BERTOT, G. GONTHIER, S. O. BIHA, I. PAŞCA. *Canonical Big Operators*, in "Proceedings of the 21st International Conference on Theorem Proving in Higher Order Logics (TPHOLs 2008)", Lecture Notes in Computer Science, Springer, August 2008, vol. 5170, pp. 12–16, http://hal.inria.fr/inria-00331193/

[4] G. GONTHIER, A. ASPERTI, J. AVIGAD, Y. BERTOT, C. COHEN, F. GARILLOT, S. LE ROUX, A. MAHBOUBI, R. O'CONNOR, S. OULD BIHA, I. PASCA, L. RIDEAU, A. SOLOVYEV, E. TASSI, L. THÉRY. *A Machine-Checked Proof of the Odd Order Theorem*, in "ITP 2013, 4th Conference on Interactive Theorem Proving", Rennes, France, S. BLAZY, C. PAULIN, D. PICHARDIE (editors), LNCS, Springer, 2013, vol. 7998, pp. 163-179 [*DOI :* 10.1007/978-3-642-39634-2_14], http://hal.inria.fr/hal-00816699

[5] G. GONTHIER, A. MAHBOUBI, L. RIDEAU, E. TASSI, L. THÉRY. *A Modular Formalisation of Finite Group Theory*, in "Proceedings of the 20th International Conference on Theorem Proving in Higher Order Logics (TPHOLs 2007)", K. SCHNEIDER, J. BRANDT (editors), LNCS, Springer-Verlag, September 2007, vol. 4732, pp. 86-101, http://hal.inria.fr/inria-00139131

Publications of the year

Doctoral Dissertations and Habilitation Theses

[6] G. CANO. *Interaction between linear algebra and analysis in formal mathematics*, Université Nice Sophia Antipolis, April 2014, https://tel.archives-ouvertes.fr/tel-00986283

Articles in International Peer-Reviewed Journals

[7] Y. BERTOT, G. ALLAIS. *Views of Pi: definition and computation*, in "Journal of Formalized Reasoning", October 2014, vol. 7, n° 1, pp. 105-129 [*DOI :* 10.6092/ISSN.1972-5787/4343], https://hal.inria.fr/hal-01074926

[8] L. FUCHS, L. THÉRY. *Implementing Geometric Algebra Products with Binary Trees*, in "Advances in Applied Clifford Algebras", February 2014, vol. 24, n° 1, 22 p. , https://hal.inria.fr/hal-01095495

[9] É. MARTIN-DOREL, G. HANROT, M. MAYERO, L. THÉRY. *Formally verified certificate checkers for hardest-to-round computation*, in "Journal of Automated Reasoning", 2015, vol. 54, n° 1, pp. 1-29 [*DOI :* 10.1007/S10817-014-9312-2], https://hal.inria.fr/hal-00919498

[10] L. THÉRY, F. WIEDIJK. *Foreword to the Special Focus on Formal Proofs for Mathematics and Computer Science*, in "International Journal of Mathematics and Computer Science", October 2014, pp. 1-3 [*DOI :* 10.1007/s11786-014-0214-9], https://hal.archives-ouvertes.fr/hal-01095761

Invited Conferences

[11] Y. BERTOT. *Links between homotopy theory and type theory*, in "CICM - Conference on Intelligent Computer Mathematics", Coimbra, Portugal, S. WATT, J. DAVENPORT, A. SEXTON, P. SOJKA, J. URBAN (editors), Springer, July 2014, https://hal.inria.fr/hal-00987248

International Conferences with Proceedings

[12] J. A. AKINYELE, G. BARTHE, B. GRÉGOIRE, B. SCHMIDT, P.-Y. STRUB. *Certified Synthesis of Efficient Batch Verifiers*, in "IEEE 27th Computer Security Foundations Symposium, CSF", Vienna, Austria, 2014 [*DOI :* 10.1109/CSF.2014.19], https://hal.inria.fr/hal-01094565

[13] G. BARTHE, F. DUPRESSOIR, P.-A. FOUQUE, B. GRÉGOIRE, J.-C. ZAPALOWICZ. *Synthesis of Fault Attacks on Cryptographic Implementations*, in "Conference on Computer and Communications Security", Scottsdale, United States, Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, ACM, 2014 [*DOI :* 10.1145/2660267.2660304], https://hal.inria.fr/hal-01094549

[14] G. BARTHE, F. DUPRESSOIR, P.-A. FOUQUE, M. TIBOUCHI, J.-C. ZAPALOWICZ, B. GRÉGOIRE. *Making RSA–PSS Provably Secure against Non-random Faults*, in "Cryptographic Hardware and Embedded Systems - CHES 2014", Busan, South Korea, CHES 2014, Springer, September 2014, vol. LNCS 8731, pp. 206 - 222 [*DOI :* 10.1007/978-3-662-44709-3_12], https://hal.inria.fr/hal-01094057

[15] G. BARTHE, C. FOURNET, B. GRÉGOIRE, P.-Y. STRUB, N. SWAMY, S. ZANELLA BEGUELIN. *Probabilistic relational verification for cryptographic implementations*, in "The 41st Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages", San Diego, United States, January 2014, https://hal.inria.fr/hal-00935743

[16] Y. BERTOT. *Fixed Precision Patterns for the Formal Verification of Mathematical Constant Approximations*, in "Certified Programs and Proofs (CPP'15)", Mumbai, India, ACM, January 2015 [*DOI :* 10.1145/2676724.2693172], https://hal.inria.fr/hal-01074927

Research Reports

[17] J. GRIMM. *Fibonacci numbers and the Stern-Brocot tree in Coq*, Inria Sophia Antipolis, December 2014, n° RR-8654, 76 p. , https://hal.inria.fr/hal-01093589

[18] J. GRIMM. *Implementation of Bourbaki's Elements of Mathematics in Coq: Part Two; Ordered Sets, Cardinals, Integers*, Inria Sophia Antipolis, 2015, n° RR-7150, 685 p. , https://hal.inria.fr/inria-00440786

Other Publications

[19] G. CANO, C. COHEN, M. DÉNÈS, A. MÖRTBERG, V. SILES. *Formalized Linear Algebra over Elementary Divisor Rings in Coq*, November 2014, https://hal.inria.fr/hal-01081908

[20] L. THÉRY. *A Formally-Proven Algorithm for 2-Sat Problems*, December 2014, https://hal.inria.fr/hal-01095538

[21] L. THÉRY. *Formally-Proven Kosaraju's algorithm*, December 2014, https://hal.archives-ouvertes.fr/hal-01095533