INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team marelle

Mathematical, Reasoning and Software

Sophia Antipolis - Méditerranée

Theme : Programs, Verification and Proofs

Table of contents

1. Team

Research Scientists

Yves Bertot [Team leader, INRIA, HdR]
Benjamin Grégoire [Research scientist INRIA]
Laurence Rideau [Research scientist INRIA]
Loïc Pottier [Research scientist INRIA, HdR]
Laurent Théry [Research scientist INRIA]

Faculty Member

Frédérique Guilhot [Qualified teacher, *académie de Nice*]

Technical Staff

Anne Pacalet
Thomas Hutchinson [ADT Coq]

PhD Students

Guillaume Cano [supervised by Y. Bertot]
Maxime Dénès [supervised by Y. Bertot]
Nicolas Julien [supervised by Y. Bertot]
Sylvain Heraud [supervised by B. Grégoire]
Sidi Ould Biha [supervised by L. Théry, until February 2010]
Santiago Zanella [supervised by Gilles Barthe, until november 2010]
Ioana Paşca [supervised by Y. Bertot, until November 2010]
Tuan Minh Pham [supervised by Y. Bertot]
Jorge Luis Sacchini [supervised by B. Grégoire]
Michaël Armand [supervised by L. Théry and B. Grégoire]

Administrative Assistant

Nathalie Bellesso [Administrative assistant]

2. Overall Objectives

2.1. Introduction

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.

Mathematical knowledge can also be made concrete in the form of decision procedures, often of the form of "satisfiability modulo theory" which can be connected to theorem proving tools in a way that preserves the trustability of the final results.

2.2. Highlights

A major result of our team this year is the IND-CCA proof of cryptographic security of RSA-OAEP, in an effort undertaken mostly by two Marelle researchers and using the Certicrypt tool developed in our team.

A second highlight of the impact of our work is the translation in Chinese of the Coq'Art book, which was a major publication of our team a few years ago.

3. Scientific Foundations

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. Second, 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). Thus, 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 also investigate the algorithms that occur when implementing programming languages, 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 of bugs.

3.4. Proof environments

We study how to improve mechanical tools for searching and verifying mathematical proofs so that they become practical for engineers and mathematicians to develop software and formal mathematical theories. There are two complementary objectives. The first is to improve the means of interaction between users and computers, so that the tools become usable by engineers, who have otherwise little interest in proof theory, and by mathematicians, who have little interest in programming or other kinds of formal constraints. The second objective is to make it easier to maintain large formal mathematical developments, so they can be re-used in a wide variety of contexts. Thus, we hope to increase the use of formal methods in software development, both by making it easier for beginners and by making it more efficient for expert users.

4. Application Domains

4.1. Certified scientific algorithms

For some applications, it is mandatory to build zero-default software. One way to reach this high level of reliability is to develop not only the program, but also a formal proof of its correctness. In the Marelle team, we are interested in certifying algorithms and programs for scientific computing. This is related to algorithms used in industry in the following respects:

- Arithmetical hardware in micro-processors,
- Arithmetical libraries in embedded software where accuracy is critical (global positioning, transportation, aeronautics),
- Verification of geometrical properties for robots (medical robotics),
- Verification of probabilities of breaking for cryptographic algorithms,
- Fault-tolerant and dependable systems.

5. New Results

5.1. Type theory and formalization of mathematics

5.1.1. Group theory

Participants: Georges Gonthier [Microsoft Research], Assia Mahboubi [project-team Typical], Laurence Rideau, Laurent Théry, Sidi Ould Biha.

We participate in the collaborative research agreement “Mathematical Components” with Microsoft Research. This project aims at evaluating the applicability of a new approach to mathematical proofs called “small-scale reflection”, especially in the domain of finite group theory [3].

This year, we have consolidated the Phd work of Sidi Ould Biha. The algebraic structures for linear algebra are now part of the main development line of the "Mathematical Components" libraries. In conjunction with some basic notions of representation theory, we have now all the pre-requisite elements for formalising the character theory that is needed for the Feit-Thompson theorem. In particular, we have included special support for finite aspects, for instance finite dimension vector spaces. This work is also supported by the Formath European project.

5.1.2. Proofs in geometry

Participants: Tuan Minh Pham, Yves Bertot.

We completed our work on developing a proving tool that integrates the capabilities of a proof system lik Coq, a proo management interface like Pcoq, and a tool for dynamic geometry manipulation and visualization like GeoGebra. This work was presented at the UITP conference.

We integrated the previous work of F. Guilhot on the formalization of high-school geometry with the work of J. Narboux on the area method for automatic proof in geometry. This also involved removing many of the axioms present in the initial work of Guilhot, where axioms were often used for definitional purposes.

Last we completed our work on describing orientation in geometry proofs.

5.1.3. Towards constructive algebraic topology

Participants: Yves Bertot, Laurence Rideau.

As part of our collaboration in the Formath European project, we gave a one week course of ssreflect at the university of La Rioja in Logroño, Spain, and we participated in the formalization of "incidence simplicial matrices" in ssreflect. We started working on an article describing this work.

5.1.4. On Bernstein coefficients

Participants: Assia Mahboubi [project-team Typical], Yves Bertot.

As a contribution to our long term objective of developing a formally verified implementation of cylindrical algebraic decomposition, we studied the proof that the number of alternation in a polynomial's Bernstein coefficients gives an upper bound of the number roots for this polynomial in the corresponding interval. An article describing this work has been submitted for publication and is already available as a pre-print [14].

5.1.5. Computing with polynomials and matrices

Participants: Maxime Dénès, Stefania Dumbrava [Bremen university], Laurent Théry.

The libraries of the project "Mathematical Components" propose a rather complete formalisation of polynomials and matrices. Unfortunately, these objects cannot be used directly for computing. In her internship, Stefania Dumbrava has been working on providing some computational contents to these objects. In particular she has investigated how persistent arrays could be effectively used for this purpose.

5.1.6. Co-recursion and real numbers

Participants: Yves Bertot, Nicolas Julien, Ioana Paşca.

The work we did on the formal verification of programs that combine Newton's method and rounding has been summarized in an article that is submitted for publication and is already available as a pre-print [15].

5.1.7. Regularity of interval matrices

Participants: Yves Bertot, Guillaume Cano, Ioana Paşca.

We have formalized a collection of criteria for the regularity of matrices with interval coefficients taken from the work of Rex and Rohn. This work leads to a publication in a conference [12] and to a chapter in Ioana Pasca's thesis [5].

The formalization relies on a theorem of mathematics whose proof has yet to be completed: the Perron-Frobenius theorem. The formal verification of this theorem is under way, it implies adding new concepts to the libraries, among which complex numbers, general topology, compacts, etc.

5.1.8. Type-based termination

Participants: Benjamin Grégoire, Jorge Luis Sacchini.

We extended the Calculus of Inductive Constructions with a type-based mechanism for ensuring termination of recursive functions. In [11] we published a preliminary version where only natural numbers are considered. We are currently working on the full version with inductive types which will be part of Jorge Luis Sacchini's Phd thesis.

5.1.9. Native compilation of terms with primitive structures

Participants: Benjamin Grégoire, Maxime Dénès.

We integrated the native compiler of the Ocaml language into a scheme for the efficient reduction of terms in the calculus of inductive constructions. On some examples, efficiency gains can reach a tenfold increase in speed. We expect this to have a strong impact on the capability to perform proofs by reflection involving heavy computation in the Coq system.

5.2. Proving tools

5.2.1. Connecting an SMT prover and Coq

Participants: Michaël Armand, Germain Faure [project-team Typical], Benjamin Grégoire, Chantal Keller [project-team Typical], Laurent Théry.

We completed our work on integrating SAT technology inside Coq. This work has been described in a publication at the conference ITP10 in Edinburgh [7]. Furthermore this serves as a basis for the integration of SMT technology. We are now capable of replaying traces produced by the SMT prover VERIT that deal with congruence closure. This works is supported by the ANR Decert project.

5.2.2. Proofs certificates for theorems in geometry

Participants: Benjamin Grégoire, Loïc Pottier, Laurent Théry.

We completed our work of previous years on Gröbner bases for geometric theorems by publishing a paper [6].

5.2.3. Geometric Algebras and Automatic Theorem Proving

Participants: Laurent Fuchs [Université de Poitiers], Laurent Théry.

We extended our formalisation of geometric algebras with some notions of bracket algebras. This lets us derive a reflexive tactic that is capable of proving elementary problems in incidence geometry fully automatically. This work was presented at the conference ADG'2010 in Munich. This work is also supported by the ANR Galapagos project.

5.2.4. A tactic on polynomial equalities: nsatz

Participant: Loïc Pottier.

We re-wrote and finished the implementation of the tactic "nsatz", which implements Hilbert's Nullstellensatz: it proves equations between polynomials from similar hypotheses. It extends the "ring" tactic. "nsatz" is implemented using the "type classes" of the Coq system, and works on integral domains, with specializations on Z, Q and R. This is available in the distributed version of the Coq system (8.3). We plan to extend this work by providing certificates in Coq for Gröbner bases, and other useful computational objects in computer algebra of this kind (dimension, invariants, etc).

5.2.5. D-Modules

Participant: Loïc Pottier.

We studied normalization of non-commutative polynomials ad exponentials in the Weyl algebra, and found a method of normalization by evaluation which reduces to the commutative case, which is suitable for an easy implementation and proof in Coq. Extension to non-commutative Gröbner bases is planned.

5.3. Formal study of cryptography

5.3.1. Certicrypt

Participants: Gilles Barthe, Yassine Lakhnech [University of Grenoble], Benjamin Grégoire, Sylvain Heraud, Santiago Zanella.

CertiCrypt is a general framework to certify the security of cryptographic primitives in the Coq proof assistant.

We have extended Certicrypt with new techniques allowing to complete the formal proof IND-CCA security of the OAEP padding schemes. The first technique is a logic for bounding the probability of an event in a game. The second technique clarifies the eager/lazy sampling methodology using a logic for swapping program statements. This work was published in [8].

We completed a machine-checked proof of the security of OAEP (a widely used public-key encryption scheme based on trapdoor permutations) security against adaptive chosen ciphertext attacks under the assumption that the underlying permutation is partial-domain one-way.

We studied Zero-knowledge proofs, which are widely applicable in cryptography, concentrating on Σ-protocols, for which we provide the first compehensive formalization in [9].

We started work on formalizing a recent proof by Icart and Coron concerning the study of hash function using elliptic curves. This work re-uses our work on Certycript and our previous work on elliptic curves.

5.3.1.1. Easycrypt

Participants: Gilles Barthe [IMDEA], Benjamin Grégoire, Sylvain Heraud, Anne Pacalet, Santiago Zanella.

Based on our experience with Certicrypt, we have started the development of the tool Easycrypt. 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 instead of using Coq. We have applied Easycrypt on a variety of academic examples and one bigger example: the proof of IND-CCA security of the Cramer-Shoup cryptosystem.

The drawback of this tool is that it provide less guarantees on the correctness of the proof than Certicrypt. To fill this gap we have started the generation of Coq file allowing to check the validity of Easycrypt proofs.

5.4. Verification of Programming tools

5.4.1. A Weakest pre-condition tool for Frama-C

Participant: Anne Pacalet.

We collaborate with the CEA to develop Frama-C which is a suite of tools dedicated to the analysis of the source code of software written in C. The 2009-2011 objective is to develop a Weakest Precondition plugin to compute proof obligations that ensures that some given properties of programs hold. The chalenge is to provide several memory model in order adapt the abstraction level of the verification. In 2010, the 2009 results have been improved to add another memory model and to transform the prototype into a more usable tool. In the middle of 2010, we managed to provide an alpha version to selected users, and the first release within Frama-C distribution is planned for December.

6. Contracts and Grants with Industry

6.1. Contracts with Industry

- We participate in the common laboratory between INRIA and Microsoft Research, in the Collaborative research actions “Mathematical components” and “Secure Distributed Computations and their Proofs”. Other participants in the first collaboration are the INRIA project-teams TYPICAL and PROVAL. The goals are to study finite group theory and efficient arithmetics. In the second collaboration, other participants are the INRIA teams INDES and MOSCOVA. We focus on formal proofs for computational Cryptography.
- We collaborate with the CEA to develop Frama-C which is a suite of tools dedicated to the analysis of the source code of software written in C.

7. Other Grants and Activities

7.1. National initiatives

- We lead the ANR project Galapagos, which started on Nov. 19th 2007. Other participants in this contract are the universities of Strasbourg and Poitiers, the ENSIEE in Evry and the Ecole Normale Supérieure in Lyon. The objective of this contract is to study the formal description of geometric concepts and algorithms.
- We participate to the ANR SCALP, which started on January 1st, 2008. Other participants in this contract are DCS-Verimag (Grenoble), Plume-LIP (Lyon), Proval-LRI (Orsay), CPR-Cédric (Cnam, Paris). In this project we focus on the formalization of Cryptography.
- We participate to the ANR project DeCert, which started on January 2009. Other participants are CEA List (Paris), LORIA-INRIA (Nancy), Celtique (IRISA Rennes), Proval (LRI Orsay), Typical (INRIA Saclay), Systerel (Aix-en-provence). The objective of the DeCert project is to design an architecture for cooperating decision procedures. To ensure trust in the architecture, the decision procedures will either be proved correct inside a proof assistant or produce proof witnesses allowing external checkers to verify the validity of their answers.
- We participate to the ANR project TAMADI, which started in October 2010. Other participants are ARENAIRE-INRIA Rhone-Alpes and the PEQUAN team from University of Paris VI Pierre and Marie Curie. The objective of the TAMADI project is to study question of precision in floating-point arithmetic and to provide formal proofs on this topic.

7.2. European initiatives

- We participate in the European project Formath, which is a STREP project in the ICT program (grant agreement number 243847). The other participants are the Universities of Göteborg (Sweden, coordinator), Nijmegen (the Netherlands), and La Rioja (Spain) and the Typical group from INRIA Saclay-Île de France. In this project, we concentrate on developing mathematical libraries for algebra, linear algebra, and algebraic topology.
- As part of the Formath project, Yves Bertot and Laurence Rideau visited the University of Logroño in Spain in June for a week, where Laurence Rideau gave a course on `ssreflect` and Yves Bertot developed a small demonstrator for algebraic topology.

7.3. International Initiatives

- Yves Bertot taught at the Asian School of Formal Methods in Beijing in August 2010.
- Sylvain Heraud spent three months at AIST in Tokyo, Japan, to collaborate with David Nowak, on formal verification of polynomial time functions used in complexity theory.

8. Dissemination

8.1. Animation of the scientific community

- Laurence Rideau was one of the organizers of the Conferences on Intelligent Computer Mathematics, which took place in Paris from July 5 to July 10, 2010. This event federated the conferences *Artificial Intelligence and Symbolic Computations*, *Calculemus*, *Mathematical Knowledge Management*, and the workshops *Compact Computer Algebra*, *Digital Mathematical Libraries*, *Mathematically Intelligent Proof Search*, *Programming Languages for Mechanized Mathematics Systems*,, *OpenMath*, *Symbolic Computation Infrastructure for Europe*.

- Yves Bertot organized a one-day Coq Workshop in July in Edinburgh [13].
- Members of the project-team participated in program committees for UITP, PAR, LFMTP, Coq-workshop, SAC-SVT, and for the Journals JAR (Journal of Automated Reasoning), JFR (Journal of Formalized Reasoning), JFP (Journal of Funcgtional Programming).
- Members of the project-team refereed papers for the conferences Calculemus, ICFP (International Conference in Functional Programming), ITP (Interactive theorem proving), JFLA (Journées Francophones des Langages Applicatifs) UITP (User-Interfaces for Theorem Provers) and for the journals JFR (Journal of Formalized Reasoning), JAR (Journal of Automated Reasoning).
- Members of the project reviewed projects for the Dutch organization for research (NWO)
- Yves Bertot was a Jury member for the theses of Christophe Brun (U. Strasbourg) and A. Charguéraud (U. Paris-Diderot).

8.2. Conference and workshop attendance, travel

Anne Pacalet attended several meetings with CEA, in Saclay, in January, February, May, September, and December.

Yves Bertot, Benjamin Grégoire, Thomas Hutchinson, and Jorge Luis Sacchini attended a Coq meeting in La Ciotat, in February.

Benjamin Grégoire visited IMDEA in Spain, in February and November, as part of his work for the SCALP ANR project.

Tuan Minh Pham attended the SAC conference in Sierre, Switzerland, in March.

Laurence Rideau and Yves Bertot attended the kick-off meeting for the Formath Workshop in Göteborg, in April.

Yves Bertot attended the Unisciel conference in Valenciennes, in May.

Ioana Pasca presented her work in Saclay, in May.

Laurence Rideau and Ioana Pasca attended the CICM event in Paris in July, and Ioana Pasca presented her work at the Calculemus conference.

Yves Bertot, Thomas Hutchinson, Tuan Minh Pham, Jorge Luis Sacchini, and Laurent Théry attended the FLOC event in Edinburgh, UK, where Yves Bertot presented work at the Coq workshop and at the ITP conference, Tuan Minh Pham presented work at UITP, and Laurent Théry presented work at ITP.

Laurent Théry and Tuan Minh Pham attended the ADG conference in Munich, in July, where they presented extended abstracts.

Laurent Théry attended a workshop on trusted extensions of theorem provers in Cambridge, UK, in August, where he gave an invited talk.

Yves Bertot gave courses in Beijing, China, in August.

Yves Bertot, Loïc Pottier, and Laurence Rideau attended the Types conference in Warsaw, in October, where Yves Bertot gave an invited talk.

Yves Bertot and Thomas Huchinson attended a meeting of the Coq working group in October.

Jorge Luis Sacchini attended the LPAR-17 conference in Yogyakarta (Indonesia), in October, where he presented work.

Laurence Rideau, Laurent Théry, Guillaume Cano, Benjamin Grégoire and Yves Bertot attended several meetings of the INRIA-Microsoft Research common laboratory, in June, October, November.

Yves Bertot, Maxime Dénès, Laurence Rideau, and Loïc Pottier attended the MAP conference in Logroño, Spain, in November.

Yves Bertot gave an invited talk at the GDR-LAC Workshop in Paris in November.

Yves Bertot visited the University of Bologna in December.

Michaël Armand and Benjamin Grégoire attended a meeting of the ANR project DECERT in Paris, in November.

8.3. Teaching

Yves Bertot *Sémantique des langages de programmation I* (Programming language semantics I), 1st year Master (18 hours), University of Nice. *Sémantique des langages de programmation, techniques avancées* (Programming language semantics, advanced techniques), 1st year Master, special cursus at University of Nice (pensionnaires de l'école normale supérieure).

Benjamin Grégoire *Vérification et sécurité*, 2nd year Master, University of Nice (18 hours).

Loïc Pottier *Logic* Engineering School *Polytech'Nice*, (38 hours), Lab sessions for the cours of programming language semantics (18 hours).

Laurent Théry *Introduction to Coq* École des Mines de Paris, (3 hours).

9. Bibliography

Major publications by the team in recent years

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

[2] 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, p. 12–16, http://hal.inria.fr/inria-00331193/.

[3] 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, p. 86-101, http://hal.inria.fr/inria-00139131.

[4] L. THÉRY. *A Machine-Checked Implementation of Buchberger's Algorithm*, in "Journal of Automated Reasoning", 2001, vol. 26, p. 107–137.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[5] I. PAŞCA. *Formal Verification for Numerical Methods*, Université Nice - Sophia Antipolis, November 2010, http://tel.archives-ouvertes.fr/tel-00555158.

Articles in International Peer-Reviewed Journal

[6] B. GRÉGOIRE, L. POTTIER, L. THÉRY. *Proof certificates for algebra and their application to automatic geometry theorem proving*, in "Lecture Notes in Artificial Intelligence", 2010, http://hal.inria.fr/inria-00504719.

International Peer-Reviewed Conference/Proceedings

[7] M. ARMAND, B. GRÉGOIRE, A. SPIWACK, L. THÉRY. *Extending Coq with Imperative Features and its Application to SAT Verification*, in "Interactive Theorem Proving", Edinburgh Royaume-Uni, Lecture Notes in Computer Science, Springer, 2010, vol. 6172, p. 83-98, http://hal.archives-ouvertes.fr/inria-00502496/en/.

[8] G. BARTHE, B. GRÉGOIRE, S. ZANELLA BÉGUELIN. *Programming Language Techniques for Cryptographic Proofs*, in "Interactive Theorem Proving, First International Conference, ITP 2010", Edinburgh, UK, M. KAUFMANN, L. C. PAULSON (editors), Lecture Notes in Computer Science, Springer, July 11-14 2010, vol. 6172, p. 115-130, http://hal.inria.fr/inria-00552894.

[9] G. BARTHE, D. HEDIN, S. ZANELLA BÉGUELIN, B. GRÉGOIRE, S. HERAUD. *A Machine-Checked Formalization of Sigma-Protocols*, in "Proceedings of the 23rd IEEE Computer Security Foundations Symposium, CSF 2010", Edinburgh, UK, IEEE Computer Society, July 17-19 2010, p. 246-260, http://hal.inria.fr/inria-00552886.

[10] J.-F. DUFOURD, Y. BERTOT. *Formal Study of Plane Delaunay Triangulation*, in "Interactive Theorem Proving, First International Conference, ITP 2010", Edinburgh, UK, M. KAUFMANN, L. C. PAULSON (editors), Lecture Notes in Computer Science, Springer, July 11-14 2010, vol. 6172, p. 211-226, http://hal.archives-ouvertes.fr/inria-00504027/.

[11] B. GRÉGOIRE, J. SACCHINI. *On Strong Normalization of the Calculus of Constructions with Type-Based Termination*, in "Proceedings 17th International Conference on Logic for Programming, Artificial Intelligence and Reasoning", Yogyakarta Indonésie, October 10-15 2010, p. 333-347, The original publication is available at www.springerlink.com [*DOI :* 10.1007/978-3-642-16242-8_24], http://hal.archives-ouvertes.fr/hal-00537104/en/.

[12] I. PAŞCA. *Formally Verified Conditions for Regularity of Interval Matrices*, in "Intelligent Computer Mathematics, 10th International Conference, AISC 2010, 17th Symposium, Calculemus 2010, and 9th International Conference, MKM 2010", Paris, France, S. AUTEXIER, J. CALMET, D. DELAHAYE, P. D. F. ION, L. RIDEAU, R. RIOBOO, A. P. SEXTON (editors), Lecture Notes in Computer Science, Springer, July 5-10 2010, vol. 6167, p. 219-233, http://hal.archives-ouvertes.fr/inria-00464937.

Books or Proceedings Editing

[13] Y. BERTOT (editor). *Second Coq Workshop*, 7 2010, http://hal.inria.fr/COQ2010.

Research Reports

[14] Y. BERTOT, F. GUILHOT, A. MAHBOUBI. *A formal study of Bernstein coefficients and polynomials*, INRIA, 7 2010, http://hal.archives-ouvertes.fr/inria-00503017/.

[15] I. PAŞCA. *Formal Proofs for Theoretical Properties of Newton's Method*, INRIA, 03 2010, n^{o} RR-7228, http://hal.inria.fr/inria-00463150/en/.