INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team typical

Types, Logic and computing

Saclay - Île-de-France

Theme : Programs, Verification and Proofs

Table of contents

The TypiCal project is a common project gathering researchers from INRIA Saclay – Île-de-France sud, École Polytechnique, and CNRS at LIX. The team leader is Benjamin Werner.

1. Team

Research Scientists

Benjamin Werner [DR INRIA, Team Leader]
Bruno Barras [CR INRIA]
Germain Faure [CR INRIA]
Assia Mahboubi [CR INRIA]

Faculty Member

Gilles Dowek [Professor, École Polytechnique, HdR]

Technical Staff

Jean-Marc Notin [IR CNRS]

PhD Students

Bruno Bernardo [ATER Paris 7 until October 2010]
Eric Biagioli [Allocation INRIA-Saclay IdF, until September 2010]
Mathieu Boespflug [AMN]
Cyril Cohen [Allocataire Ministère]
Chantal Keller [ENS Lyon]
Victor Magron [Projet européen FORMATH, since October 2010]
Pierre Néron [Allocataire Ministère]
Arnaud Spiwack [ENS Cachan]

Administrative Assistant

Valérie Lecomte [TR INRIA]

2. Overall Objectives

2.1. Presentation

Mathematics is among the many human activities that have been transformed by the invention of the computer and its broad diffusion in the second half of the 20^{th} century. Mathematicians could, from then on, use a tool allowing to carry out operations that were too long or too tedious to be executed by hand. Like the use of the telescope in astronomy, the use of the computer opened many new prospects in mathematics. One of these prospects is the use of *proof assistants*, *i.e.* computer programs which perform some operations on mathematical proofs. The goal of the research developed in the TypiCal project-team is to develop such *proof assistants*. The main effort of the project-team is to contribute to the development proof assistants in general and of the Coq system in particular, which has an important community of users in industry and in academia. However, we believe that the development of a proof assistant cannot be accomplished without a joint reflection about the structure of mathematical proofs and about the use of proof assistants in various applicative domains. We also believe that proof assistants should take benefit of the use of automated deduction tools. Thus, the questions addressed in the team range from questions related to the Coq system, such as "What will be the features of the next version of Coq?", to more theoretical questions of logic, such as "What is a proof?" and more applied ones, such as "How can I delegate part of the proof search to automated tools?" or "How can we use a proof assistant to check whether a protocol is free of deadlocks?".

3. Scientific Foundations

3.1. Proof assistants

The first operation that a proof assistant can perform on a proof is to check its correctness. This participates in the quest for a new step in mathematical rigor: the point where nothing is understated, and where the reader can therefore be replaced by a program. This quest for rigor is specially important for the large proofs, either hand written or computer aided, that mathematicians have built since the middle of the 20^{th} century. For instance, without using a proof assistant, it is quite difficult to establish the correctness of a proof using symbolic computations on polynomials formed with hundreds of monomials, or a case analysis requiring the inspection of several hundreds of cases, or establishing that a complex object such as a long program or a complex digital circuit has some property. This quest for correctness is especially important in application domains where a malfunction may jeopardize human life, health or environment, such as transportations or computer aided surgery.

Besides this correctness check, proof assistants can help the users to build proofs interactively. The "tactic language" allowing the user to control the system in this proof construction process has always been the object of intensive studies. The ML language, for instance, was originally the tactic language of the LCF proof assistant. More recent questions about this language are focused on the formal expression of its operational semantic, in particular the handling of exceptions.

Proof assistants may also prove some easy but big size lemmas automatically. The automatisation of proof assistants can be increased by the development of decision procedures. Either they can be developed inside the proof assistant or we can use external tools producing certificates later used inside the proof assistant to reconstruct the proof.

Proof assistants may also transform mathematical proofs into other formal objects such as programs.

A more recent kind of applications is the construction of large libraries of mathematical results on the net.

3.2. Formalization of mathematics

A proof assistant implements a particular formalism allowing to express mathematics. A traditional formalism allowing to express mathematics is set theory, built on top of first-order predicate logic. Unfortunately, this formalism does not address exactly the needs of a proof assistant. Set theory has been elaborated at the beginning of the 20^{th} century to study mathematically the properties of mathematical reasoning. For this purpose, being able to formalize mathematics "in principle" was enough. Nowadays, the problem is not to formalize mathematics "in principle" but to formalize them "in facts". Thus, the design of proof assistants has led to ask new questions in logic and, in particular, in proof theory.

Several variants or alternative to set theory have been designed to express mathematics in practice. The system Coq is based on a formalism called *The Calculus of Inductive Constructions*.

An important feature for such a formalism is the language allowing to express mathematical objects such as functions and sets. It is not desirable to use a formalization of mathematics that has only existence axioms, or even one having the combinator's language obtained by skolemizing these axioms in predicate logic. It is important to have a rich and compact language, in particular a language with binders such as the λ-calculus.

Another important feature is the ability to integrate deduction and computation. It is not possible, when we use a proof assistant to consider that the proposition $2 + 2 = 4$ requires a proof, even a proof simple enough to be found by a automated theorem proving system. Several formalisms such as Martin-Löf's type theory, Boyer-Moore logic, the Calculus of Constructions and the Calculus of Inductive Constructions, include such a possibility to compute inside a proof. Thus, these formalisms designed to express mathematics contain a programming language as a sub-language.

More recently the research in this area has taken several different directions: first the study of *deduction modulo* that is the simplest extension of predicate logic allowing to mix deduction and computation. Deduction modulo has applications both in automated theorem proving and in proof theory, where it paves the way to a unified theory of cut elimination. Finally, the need to improve the efficiency of computations in the system Coq, has led to the use of compilation techniques issued from the theory of programming language. This has brought logical languages and programming languages closer, allowing for instance to use the language of Coq as a general purpose programming language. This perspective of unifying proof and programming languages is a real challenge for future proof assistants.

Another property of the Calculus of Inductive Constructions that is important for its use as the language of a proof assistant is the possibility to write both constructive and classical proofs. When a proof of existence is constructive, the user can request the computation of a witness, but, of course, not when it is classical.

By insisting on this idea that *constructive proofs* must be distinguished from classical proofs, the project-team TypiCal participates to rise of a new form a constructivism, not trying to restrict mathematics to constructive mathematics, but trying to identify the part of mathematics that can be done constructively and the part that cannot.

A last property of the Calculus of Inductive Constructions is that proofs are objects of the formalism, exactly as numbers, functions and sets are. This property, based on the celebrated Curry-De Bruijn-Howard correspondence, allows to reduce the safety critical base of the Coq system to a quite small kernel.

4. Application Domains

4.1. Application Domains

The first application is to pure mathematics. The use of proof assistants for proving genuine mathematical theorems has been considered as utopic for long. But several recent developments have changed the situation. First of all, the development of libraries of both constructive and classical analysis has led the possibility to use Coq, not only in remote areas of discrete mathematics, but also to prove mainstream mathematical theorem as taught in an undergrad textbook for instance. This direction culminated with the proof in Coq of the Fundamental Theorem of Algebra, a few years ago, by a group of researchers in Nijmegen. More recent work include a proof of the Four color theorem in Coq, proofs of lemma's on polynomials used in the proof of Hale's Sphere packing theorem (Kepler's conjecture), proofs in algebraic geometry by a group of mathematicians in Nice. The Mathematical Components group of the INRIA - MSR Joint Centre is working on the formalisation of the Feit Thompson theorem (1962) for groups of odd order, which is a milestone in the classification of finite groups.

Another direction is the proof of algorithms. In proofs of algorithms (as opposed to proofs of programs) a property is proved on an algorithms formalized in the language of Coq. An example is the recent proof of algorithms used in floating point arithmetic or the older proof carried out by the company *Trusted Logic* of the correctness that has reached, for the first time, the EAL7 level in common criteria.

The most applied use of Coq is the proof of programs where an actual program written in the syntax of a general purpose programming language (such as Caml, Java or C). The system Coq is used by the ProVal project-team, that has strong historical connections to TypiCal, as a back-end of their systems Why, Krakatoa and Caduceus.

5. Software

5.1. Coq

The TypiCal team participates to the developments of the *Coq* system. The *Coq* system is a processor of mathematical proofs allowing an interactive development of specifications and proofs.

At the architectural level, the main feature is the isolation of the critical code preforming the proof checking in a kernel small enough to reach higher levels of reliability of the whole system (with the current goal of achieving the self-validation), and the production of an abstract interface of that kernel granting that theories can only be built using the features of the kernel. A standalone checker of compiled libraries can be used to validate libraries with an even higher level of confidence.

Coq is used in hundreds of sites. We have demanding users in industry (France Télécom R & D, Dassault-Aviation, Trusted Logic, Gemplus, Schlumberger-Sema, ...) in the academic world in Europe (Scotland, Netherlands, Spain, Italy, Portugal, ...) and in France (Bordeaux, Lyon, Marseille, Nancy, Nantes, Nice, Paris, Strasbourg, ...).

The *Coq* system is available from URL http://coq.inria.fr/. Written in Objective Caml and Camlp4, it is ported to most Unix architectures, but also to Windows and MacOS.

5.2. Dedukti

A universal proof checker called DEDUKTI has been realeased. This is an implementation of a type checker for proofs written in the $\lambda\Pi$-modulo calculus.

To make the translator useful in practice, we are currently writting different translator from already existing proof assistants to DEDUKTI. Our effort are now concentrated on the translation from Coq. This raised difficult theoretical questions that are under investigation.

DEDUKTI is available from ULR http://www.lix.polytechnique.fr/dedukti. It is written in HASKELL and is short enough to be trustable. It is a product of a long run collaboration with other INRIA teams. This collaboration was formalized by an INRIA ARC (special funding for collaborated INRIA teams).

6. New Results

6.1. Development of theories and tactics

6.1.1. Translation of HOL Light proofs into Dedukti

Participants: Mathieu Boespflug, Germain Faure, Chantal Keller, Benjamin Werner.

This is work in collaboration with Paul Brauner. One of Dedukti's aims is to check proofs coming from different systems. A translator from Higher Order Logic into $\lambda\Pi$-modulo has been studied, and implemented to translate HOL Light proofs into Dedukti. It reuses some ideas of [kellerW10]. The adequacy of the translation remains to be proved.

6.1.2. A normalizer for the simply-typed λ-calculus in Total Type Theory

Participant: Chantal Keller.

This is work in collaboration with Thorsten Altenkirch. In Total Type Theory, all functions must terminate. Interactive theorem provers implementing it, such as Agda, usually check termination using structural arguments. It is thus not obvious how to write a normalizer for the λ-calculus, since common normalizers are not structurally recursive. We implemented a normalizer for the simply typed λ-calculus in Agda using the hereditary substitutions algorithm, which is structurally recursive. We then showed that this normalizer can be used to decide $\beta\eta$-equality.

6.1.3. Bernstein polynomials

Participant: Assia Mahboubi.

Yves Bertot and Assia Mahboubi have completed a Coq library formalizing constructive proofs of the fundamental properties of Bernstein polynomials. This work is upgrading and extending a preliminary study they had started in collaboration with Fréderique Guilhot in 2006. Bernstein polynomials provide a discrete approximation of polynomials inside a bounded interval. As such they are useful tools to solve problems like locating the roots of polynomials, isolating these roots or solving systems of inequations with polynomial members. They are pervasive in computer aided design but they can also constitute a fundamental ingredient in the implementation of the Cylindrical Algebraic Decomposition algorithm.

6.1.4. First-order linear arithmetic in the Kepler conjecture

Participant: Assia Mahboubi.

Assia Mahboubi has supervised the 3rd year internship of Nathaniel Carré (École Polytechnique). They have worked on the production of formal proofs for the linear systems occuring in Thomas Hales' proof of the Kepler conjecture. These unfeasible linear problems have been originally dealt with using specialized operational research tools and most of them have been formally proved infeasible using the Isabelle proof assistant. The aim of this work is to take benefit of the computation abilities of the Coq proof assistant to obtain a formal proof for all systems, with a smaller formal certificate. This is still work in progress.

6.1.5. Decision procedures for integer linear arithmetic

Participant: Assia Mahboubi.

In collaboration with members of the ProVal team (Sylvain Conchon,Évelyne Contejean, Mohammed Iguernelala and Alain Mebsout) and in the context of the DeCert ANR project, Assia Mahboubi has studied a new decision procedure for integer arithmetic, implemented in the Alt-Ergo SMT solver. This procedure is based on a better cooperation with procedure already available in an SMT tool, namely interavl arithmetic and rational linear arithmetic.

6.1.6. Cylindrical Algebraic Decomposition algorithm

Participant: Assia Mahboubi.

During her stay at Stanford Research Institue, Assia Mahboubi has achieved an implementation of the Cylindrical Algebraic Decomposition algorithm in Objective Caml. This implementation is purely functional and will serve as a basis for its certification in the Coq proof assistant, in the context of the FORMATH project.

6.1.7. Formal proof for Weak memory models

Participant: Assia Mahboubi.

Assia Mahboubi has collaborated with Jade Alglave to provide a formalization in Coq of the semantic proposed by Jade Alglave in a PhD for weak memory models.

6.1.8. Quantifier elimination in algebraically closed fields

Participants: Cyril Cohen, Assia Mahboubi.

Cyril Cohen and Assia Mahboubi have implemented and certified a quantifier elimination procedure for the theory of discrete algebraically closed fields. This work uses the Ssreflect Coq extension and is based on the hierarchy of algebraic structure and a library for polynomials developed by the Mathematical Component project. This led to a publication in Calculemus conference [19].

6.1.9. Ordered ring library

Participant: Cyril Cohen.

Cyril Cohen developed a Ssreflect library about discrete ordered and partially ordered integral domains and fields. It is based on the hierarchy of algebraic structure. This is a required basis for works on polynomial analysis or for the formalization of real closed fields.

6.1.10. Quantifier elimination in real closed field

Participants: Cyril Cohen, Assia Mahboubi.

Cyril Cohen and Assia Mahboubi are working on a certified quantifier elimination procedure for the theory of discrete real closed fields. This work uses the ordered ring library to define real closed fields and to develop some polynomial real analysis required to certify the procedure. It also reuses ideas from the Quantifier elimination in algebraically closed fields. This work is still in progress.

6.1.11. Multinomials

Participant: Cyril Cohen.

Cyril Cohen has worked on a Ssreflect library for polynomials with countable indeterminate. This work could lead, for example, to development about elementary symmetric functions. It is based on the development about quotient types.

6.1.12. Effective Gaussian elimination procedure

Participant: Cyril Cohen.

Cyril Cohen has worked in collaboration with Anders Mörtberg (University of Gottenburg). This comes from a collaboration inside the FORMATH European project. This work lead to a certified and effective implementation of a Gaussian elimination procedure. This was done using tools from the Mathematical Components project.

6.1.13. Semantics of the Calculus of Inductive Constructions

Participant: Bruno Barras.

Bruno Barras has followed his development of a formal set-theoretical model of the Calculus of Inductive Constructions. He has shown that it was possible to build and prove sound a model of the Calculus of Constructions extended with the type of natural numbers seen as an inductive type in IZF. This result improves the previous ones by avoiding the use of excluded-middle and the axiom of choice.

Bruno Barras has also made a significant step towards the formalization of inductive types in the general case by producing a model of the Brouwer's ordinal inductive type (`ord`). This has been carried out assuming a well-known fact in cardinal theory, but it remains open if this can be proved constructively, i.e. without the axiom of choice.

6.2. Developments of systems

6.2.1. Release of Coq V8.3

Participant: Bruno Barras.

Bruno Barras has participated to the release of Coq V8.3 in October 2010.

6.2.2. Interfacing Coq with SMT solvers

Participants: Germain Faure, Chantal Keller, Assia Mahboubi, Benjamin Werner.

This work is in close collaboration with the Marelle team (INRIA Sophia Antipolis). The starting point of this work is to note that SMT solvers, deciding the Satisfiability Modulo Theories, are in constant evolution to take into account new decision procedures as well as theories. These systems are rather complex and it is now clearly established that they all contain bugs. The standard approach is to ask the SMT solver to append to the decision result a certificate that can be checked by another tool.

In this context, we are using formal systems like Coq to check the certificate. We are now able to check certificates coming from the SMT solver VeriT in short time, for the theory of congruence closure. We also use certificates to build a new Coq tactic that can safely call an external SMT solver, thus increasing Coq's automation.

6.3. Study of Formalisms

6.3.1. Deduction modulo

Participant: Gilles Dowek.

Gilles Dowek has revisited the resolution method in Deduction modulo by showing how this method simplifies when considering clausal rewrite rules. A publication at the workshop IFIP Theoretical Computer Science resulted from this work.

6.3.2. Computability in a physical framework

6.3.3. Deduction modulo

Participant: Gilles Dowek.

Gilles Dowek has established a relation of entailment between the pysical form of Church's thesis and Galileo's thesis. He presented an invited talk to the conference *Physics and computation* and a publication has been submitted.

Gilles Dowek and Pablo Arrighi have shown the stability of the notion of algebraic computability in many spaces studied in quantum theory. This work has been presented at the conference *Computability in Europe* and has been published in the proceedings of this conference.

Gilles Dowek and Pablo Arrighi have shown how adding axioms to quantum theory allows to prove the physical form of Church's thesis. This work has been submitted to publication.

6.3.4. Variable binding

Participant: Gilles Dowek.

Gilles Dowek and Jamie Gabbay have introduced a new logic, the permissive nominal logic, that can express theories in languages with bound variables. This work has been presented at the conference *Principles and Practice of Declarative Programming* and has been published in the proceedings of this conference. Then, they showed that higher-order abstract syntax could be explained as a translation from this logic to higher-order logic. This work is submitted to publication.

Gilles Dowek and Ying Jiang have proposed a variant of typed lambda-calculus without variables and they have shown that considering dependent types, this calculus was as expressive as lambda-calculus.

6.3.5. Closure analysis of lambda-terms

Participant: Bruno Barras.

Bruno Barras has made a proposal to improve the ability to develop programs using dependent types by splitting the dependent pattern-matching of the Calculus of Inductive Constructions into a non-dependent pattern-matching operator and a rewrite operator.

To aliviate the burden of proving the equalities that have to be given to the rewrite operator, the proposal suggests that such proofs can be omitted when they fall into a given decidable fragment. When this is not the case, the user should be allowed to provide a full proof-term. In order to remain conservative with respect to the original formalism, it is necessary to check that this proof does not rely on unprovable facts. This can be checked by a static analysis on the proof to decide if it reduces to reflexivity, provided that the equations produced by every branch of the case-analysis are proved by reflexivity.

Bruno Barras has started to develop a small auxiliary type systems that performs this analysis. It is much more precise than just checking that proof contains no free variable, since some parts of the proof might not be used to produce the reflexivity contructor.

6.3.6. Higher-order matching

Participant: Germain Faure.

Germain Faure studied higher-order matching in an untyped setting while the standard approach uses typed setting. He showed that this is particularly interesting because (1) an easy and efficient algorithm can be build (2) second-order matching is subsumed by the problems we deal with. He also showed that these results can be applied with success in the context of higher-order rewriting. This paper is accepted for publication in the Logical Methods in Computer Science international journal.

6.3.7. Extensions of the Implicit Calculus of Constructions

Participants: Bruno Barras, Bruno Bernardo.

Bruno Bernardo and Bruno Barras are working on an Implicit Calculus of Constructions with dependent sums (also known as Σ-types) and with decidable type inference. In this calculus all the static information (types and proof objects), though it appears explicitly, is transparent and does not affect the computational behavior. Bruno Bernardo and Bruno Barras have already defined and studied an Implicit Calculus of Constructions with decidable typing [33]. Next step is to add Σ-types to the system. The syntax has already been extended. Subject reduction has been proven. The extension of Alexandre Miquel's models based on coherence spaces [8] is ongoing work that would lead to prove the consistency and the strong normalisation property of the system.

7. Contracts and Grants with Industry

7.1. INRIA Microsoft Research Joint Centre

TypiCal has a strong link with the INRIA-Microsoft Research joint centre, of which Benjamin Werner, Assia Mahboubi, Cyril Cohen, and Bruno Barras are also members.

7.2. ADT Coq

TypiCal, through its participation to the development of Coq is part of the ADT (Action de Développement Logiciel) Coq. It is a specific founding by INRIA. It involves people and teams that collaborate to the implementation of the Coq proof assistant. The involved teams are the following: TypiCal, ProVal, Marelle, and πr^2 from INRIA as well as the CPR team from CNAM.

7.3. Digitéo PASO

The PASO project *(Preuves, Interprétation abstraite, and Optimisation* cal properties of programs, arising in particular from the modeling of complex systems with critical security issues. It gathers computer scientists from CEA-LIST/MeASI, INRIA Saclay/Typical & LIX and specialists from Optimisation or Control theory from LIX/MeASI, INRIA Saclay/Maxplus & CMAP, and Supelec/L2S. The goal of this exploratory project is to cross-fertilise these fields, by applying advanced algorithms or techniques inspired by global optimization, by the analysis and identification of dynamical systems, or by zero-sum game theory, in order to improve the precision or the scalability of current methods in proof and static analysis. These applications coming from computer science turn out to raise new challenges for the applied mathematicians.

7.4. ANR Decert

Assia Mahboubi and Germain Faure are part of the ANR Decert *Décision certifiée* coordinated by Thomas Jensen in Rennes. The objective of the DECERT project is to design an architecture for cooperating decision procedures, with a particular emphasis on fragments of arithmetic, including bounded and unbounded arithmetic over the integers and the reals, and on their combination with other theories for data structures such as lists, arrays or sets. 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.

7.5. ANR PSI

Assia Mahboubi and Germain Faure are part of the ANR PSI *Proof Search Interaction* coordinated by Stéphane Lengrand. The goal of the project is to understand how we can take into account a specific theory when elaborating proof search strategies, both at the level of proof theory and at the design of automated tools.

7.6. ARC Corias

Germain Faure (coordinator), Lisa Allali, Denis Cousineau, Gilles Dowek, Mathieu Boespflug are members of the ARC Corias Conception et réalisation d'assistants à la preuve basés sur la super-déduction in collaboration with the Pareo Team (INRIA Nancy Grand Est).

The project focussed on the development of an universal proof checker called Dedukti. It was based on the application of principles of (super)deduction modulo to type theory. The development of this tool requires (1) to obtain theoritical results for examples for the encoding of proofs from Coq (Calculus of Inductive Constructions formalism) in Dedukti (Lambda-calculus modulo formalism) (2) to develop implementation techniques for a successful scale-up.

The two-year project coordinated by Germain Faure ends up at the end of 2010. The overall work leads to 19 publications and the GPL code is available on line on the website of the project (see http://www.lix.polytechnique.fr/corias/). The results of the project were successfully presented during the "journées nationales des arc, adt et actions exploratoires" at INRIA Paris-Rocquencourt.

7.7. European action: FORMATH

The FET-Open European project FORMATH about formalizing mathematics lead by Thierry Coquand in Göteborg has started in March 2010. It comprises researchers of INRIA, KUN Nijmegen, La Rioja, and Microsoft Research. Assia Mahboubi and Benjamin Werner are participating for TypiCal. Assia Mahboubi is coordinating the INRIA/Paris Area subsite of the INRIA component.

8. Dissemination

8.1. Animation of the scientific community

8.1.1. Organisation of Conferences and Workshops

Assia Mahboubi is the vice-president and hence co-organizer of the national conference JFLA 2011.

8.1.2. Editorial charges

Gilles Dowek has participated to the program committees of the conferences LICS, ICALP, IJCAR, RTA, LSFA, and NFM.

Assia Mahboubi has served in the program committee of international conference CICM 2010 and ITP 2010.

Assia Mahboubi has served in the program committee of national conference JFLA 2010.

8.1.3. Committees

Assia Mahboubi has served in the selection committee of a "Maître de conférence" position in computer sciences at University of Évry/ENSIIE.

Assia Mahboubi has served in the selection committee of the Qualcomm/Lix postodctoral position at LIX.

8.1.4. Referees

Assia Mahboubi has served as a reviewer for the international conferences ITP 2010, CICM 2010 and LICS 2010.

Bruno Barras has seved as a reviewer for the Journal of Automated Reasoning and the conferences LICS 2010 and MPC 2010.

Germain Faure served as refeere for the CAV 2010 international conference, for the ITP 2010 international conference, STACS 2011 international conference.

Germain Faure and Chantal Keller served as a referee for the LPAR 2010 international conference and the post proceedings of the VSTTE2009 international workshop.

Cyril Cohen has served as referee for the ITP 2010 conference.

8.1.5. Conferences

Gilles Dowek has participated to the conferences *Computability in Europe* (Portugal), where he gave a talk, *Principles and Practice of Declarative Programming* (Austria), where ha gave a talk, *Physics and Computation* (Egypt), where he gave an invited talk, *IFIP Theoretical Computer Science* (Australie), where he gave a talk.

Gilles Dowek also participated to the conference *Philosophy of the Information and Computing Sciences* (Netherlands) where ha gave an invited talk. He has given a talk to the seminar common to CHSPAM – REHSEIS of SPHERE Histoire and "Philosophie des mathématiques" and another talk at the seminar "Philosophie et Mathématiques".

Assia Mahboubi has participated and given a talk at the Kick Off meeting of the FORMATH project in Gothenburg (Sweden) in April 2010.

Assia Mahboubi has participated to the CICM 2010 conference in Paris (France) in July 2010.

Assia Mahboubi has participated and given an invited talk at the IWS 10 workshop, part of the FLOC 2010 conference in Edimburgh (UK) in July 2010.

Assia Mahboubi has participated and given a talk at the TYPES 2010 workshop in Warsaw (Poland) in October 2010.

Assia Mahboubi has participated and given an invited course at the MAP 2010 conference in Logroño (Spain) in November 2010.

Assia Mahboubi has participated and given a talk at the ANR DeCert meeting in Paris (France) in December 2010.

Assia Mahboubi has taught two courses ans supervised the daily practical sessions at the INRIA-EDF-CEA Spring School "Modélisation et vérification d'algorithmes en Coq : une introduction" http://www.inria.fr/centre-de-recherche-inria/paris-rocquencourt/agenda/ecoles-cea-edf-inria-modelisation-et-verification-d-algorithmes-en-coq-une-introduction.

Bruno Barras has participated and given a talk at two ADT working groups in February (La Ciotat) and in October (Paris).

Germain Faure has participated and given a talk at the following events or conferences:

- ARC, ADT, and AEx 2010 day organised by INRIA, Rocquencourt (France).
- the ANR Decert Décision certifiée meeting, Paris, France.

Chantal Keller participates to the following working groups:

- ANR Decert
- ANR Psi
- ARC Corias
- GT LAC (GDR IM, CNRS)

Chantal Keller has participated and given a talk at the following events or conferences:

- the ITP'10 international conference
- the MSFP'10 international workshop
- the meetings of the working group LAC of the GDR IM (CNRS) of 3/15/10 and 11/16/10

She attended the ICFP'10 international conference and the OPLSS'10 Oregon Summer School.

Cyril Cohen has participated to and has given a talk at JFLA2010 (La Ciotat, France), at Calcumemus 2010 (Paris) and at TYPES 2010 (Varsaw, Poland). He also participated to the conference MAP 2010 (Logono, Espagne).

Arnaud Spiwack participated to and given a talk at the following conferences:

- ITP'10, part of the FLOC'10 event, in Edinburgh (UK)
- PSTT'10 workshop, part of the FLOC'10 event, in Edinburgh (UK)

Arnaud Spiwack attended the workshop PLMMS'10, part of the CICM'10 event in Paris (France).

8.1.6. Visits

Assia Mahboubi has visited Thomas Braibant and Damien Pous at INRIA Grenoble in september 2010.

Cyril Cohen has been to Gothenburg university (Sweden) to collaborate with Thierry Coquand for the FORMATH european project.

Gilles Dowek has invited Jamie Gabbay, as a Digiteo invited professor for a period of 3 months.

8.1.7. Popular science

Germain Faure has given a popular science talk at "Les olympiades mathématiques".

Gilles Dowek has given a talk at the public library of Plessis-Trévise (January 16th), at Petite Université Libre of Tence (January 23rd), and Tables Ronde La Recherche (Collège de France, June 12th) and at Institut Bull (October 21st).

Gilles Dowek has also given talks at the mathematical colloquiums of Strasbourg university (January 28th) and Poitier university (September 30th) and at École Centrale (November 3rd).

8.1.8. Other charges

Assia Mahboubi is elected as a representant of researchers at the Comité de Centre INRIA Saclay Ile de France.

Assia Mahboubi is elected as a representant of researchers at the Conseil de Laboratoire of the LIX Laboratory.

Assia Mahboubi has served as a representant of researchers at the Comité Enseignement Recherche of the Computer Sciences Department of École Polytechnique until September 2010.

Assia Mahboubi is in charge of the seminar of the TypiCal team.

Bruno Barras is consultant in formal methods at Trusted Labs, located in Versailles.

Germain Faure was the coordinator of the INRIA ARC Corias that involves teams from INRIA Nancy Grand-Est, Bordeaux Grand-Ouest, and Saclay Île-de-France.

Germain Faure has presented the LIX laboratory to the incoming students of the École Polytechnique in 2010.

8.2. Teaching

Assia Mahboubi and Benjamin Werner coadvise the thesis of Cyril Cohen. Benjamin Werner and Germain Faure coadvise the thesis of Chantal Keller. Benjamin Werner is thesis co-adviser of Arnaud Spiwack, Eric Biagoli, and François Garillot. Gilles Dowek and Bruno Barras are the thesis advisors of Bruno Bernardo. Bruno Barras coadvises the thesis of Vincent Silès. Gilles Dowek is the thesis advisor of Mathieu Boespflug and Pierre Néron.

Gilles Dowek has supervised the internship (M1) of Alexis Dorra, and the internships (M2) of Maxime Dénès and Aloïs Brunel. He also greeted Jianhua Gao who spends in France one year of his thesis at Institute of Software of the Science Academy of China.

Gilles Dowek has participated to the HDR jury of Pierre Valarcher, Fédéric Prost and David Delahaye. He also participated to the Ph.D. thesis jury of Éric Jeager, Christiphe Calvès (London), Clément Houtmann, Paul Brauner, Yvan Noyer, Danko Illik and Benoît Montagu.

Bruno Barras and Benjamin Werner teach at the *Master Parisien de Recherche en Informatique*.

Germain Faure teaches at École Polytechnique.

This year, Chantal Keller has taught a course about Proofs of programs at École Normale Supérieure des Techniques Avancées (ENSTA).

Bruno Bernardo was teaching as an ATER in Paris 7 until end of August 2010.

Cyril Cohen has been teaching at École Polytechnique as a "moniteur".

9. Bibliography

Major publications by the team in recent years

[1] G. DOWEK. *Les Métamorphoses du Calcul*, Le Pommier, 2007.

[2] G. DOWEK, O. HERMANT. *A Simple Proof That Super-Consistency Implies Cut Elimination*, in "Term Rewriting and Applications, 18th International Conference, RTA", F. BAADER (editor), Lecture Notes in Computer Science, Springer, 2007, vol. 4533, p. 93-106.

[3] G. DOWEK, B. WERNER. *Proof normalization modulo*, in "J. Symb. Log.", 2003, vol. 68, n^{o} 4, p. 1289-1316.

[4] G. GONTHIER, A. MAHBOUBI, L. RIDEAU, E. TASSI, L. THÉRY. *A Modular Formalisation of Finite Group Theory*, in "Theorem Proving in Higher Order Logics, 20th International Conference", K. SCHNEIDER, J. BRANDT (editors), Lecture Notes in Computer Science, Springer, 2007, vol. 4732, p. 86-101.

[5] B. GRÉGOIRE, L. THÉRY, B. WERNER. *A Computational Approach to Pocklington Certificates in Type Theory*, in "Functional and Logic Programming, 8th International Symposium", M. HAGIYA, P. WADLER (editors), Lecture Notes in Computer Science, Springer, 2006, vol. 3945, p. 97-113.

[6] H. HERBELIN, S. GHILEZAN. *An approach to call-by-name delimited continuations*, in "Proceedings of the 35th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL", G. C. NECULA, P. WADLER (editors), ACM, 2008, p. 383-394.

[7] H. HERBELIN. *C'est maintenant qu'on calcule, au cœur de la dualité*, Université Paris-Sud, 2005, Habilitation à diriger des recherches.

[8] A. MIQUEL. *Le Calcul des Constructions Implicites : syntaxe et sémantique*, Université Paris VII, 2001.

[9] J. NARBOUX. *Mechanical Theorem Proving in Tarski's Geometry*, in "Automated Deduction in Geometry, 6th International Workshop, ADG", F. BOTANA, T. RECIO (editors), Lecture Notes in Computer Science, Springer, 2006, vol. 4869, p. 139-156.

[10] R. ZUMKELLER. *Formal Global Optimisation with Taylor Models*, in "Automated Reasoning, Third International Joint Conference, IJCAR", U. FURBACH, N. SHANKAR (editors), Lecture Notes in Computer Science, Springer, 2006, vol. 4130, p. 408-422.

Publications of the year

Articles in International Peer-Reviewed Journal

[11] B. BARRAS. *Sets in Coq, Coq in Sets*, in "Journal of Formalized Reasoning", 2010, vol. 3, n^{o} 1, p. 29–48.

[12] Y. BERTOT, G. FRÉDÉRIQUE, A. MAHBOUBI. *A formal study of Bernstein coefficients and polynomials*, in "Mathematical Structures in Computer Science", 2010, http://hal.inria.fr/inria-00503017/en/.

[13] G. DOWEK, M. J. GABBAY, D. P. MULLIGAN. *Permissive nominal terms and their unification: an infinite, co-infinite approach to nominal techniques*, in "Logic Journal of the IGPL", 2010, vol. 18, n^{o} 6, p. 769-822.

[14] G. GONTHIER, A. MAHBOUBI. *An introduction to small scale reflection in Coq*, in "Journal of Formalized Reasoning", 2010, vol. 3, p. 95-152, http://hal.inria.fr/inria-00515548/en/.

International Peer-Reviewed Conference/Proceedings

[15] M. ARMAND, B. GRÉGOIRE, A. SPIWACK, L. THÉRY. *Extending Coq with Imperative Features and its Application to SAT Verification*, in "Interactive Theorem Proving", Royaume-Uni Edinburgh, 2010, This work was supported in part by the french ANR DECERT initiative, http://hal.inria.fr/inria-00502496.

[16] P. ARRIGHI, G. DOWEK. *On the Completeness of Quantum Computation Models*, in "Programs, Proofs, Processes, 6th Conference on Computability in Europe", F. FERREIRA, B. LÖWE, E. MAYORDOMO, L. M. GOMES (editors), Lecture Notes in Computer Science, Springer, 2010, vol. 6158, p. 21-30.

[17] M. BOESPFLUG. *Conversion by Evaluation*, in "Twelfth International Symposium on Practical Aspects of Declarative Languages", Espagne Madrid, Jan 2010, http://hal.inria.fr/inria-00434282.

[18] M. BOESPFLUG. *Conversion by Evaluation*, in "Twelfth International Symposium on Practical Aspects of Declarative Languages", Madrid Espagne, 01 2010, http://hal.inria.fr/inria-00434282/en/.

[19] C. COHEN, A. MAHBOUBI. *A formal quantifier elimination for algebraically closed fields*, in "Symposium on the Integration of Symbolic Computation and Mechanised Reasoning, Calculemus", France Paris, Springer, Jun 2010, vol. 6167, p. 189-203, http://hal.inria.fr/inria-00464237.

[20] G. DOWEK. *Polarized Resolution Modulo*, in "Theoretical Computer Science - 6th IFIP TC 1/WG 2.2 International Conference", C. S. CALUDE, V. SASSONE (editors), IFIP, Springer, 2010, vol. 323, p. 182-196.

[21] G. DOWEK, M. J. GABBAY. *Permissive-nominal logic*, in "Proceedings of the 12th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming", T. KUTSIA, W. SCHREINER, M. FERNÁNDEZ (editors), ACM, 2010, p. 165-176.

[22] C. KELLER, T. ALTENKIRCH. *Hereditary Substitutions for Simple Types, Formalized*, in "Mathematically Structured Functional Programming 2010", États-Unis Baltimore, ACM Press, 2010, http://hal.inria.fr/inria-00520606.

[23] C. KELLER, B. WERNER. *Importing HOL Light into Coq*, in "Interactive Theorem Proving", Royaume-Uni Edimbourg, Springer, 2010, vol. 6172, p. 307-322, http://hal.inria.fr/inria-00520604.

[24] A. SPIWACK. *An abstract type for constructing tactics in Coq*, in "Proof Search in Type Theory", Royaume-Uni Edinburgh, 2010, http://hal.inria.fr/inria-00502500.

Scientific Books (or Scientific Book chapters)

[25] G. DOWEK. *Les démonstrations et les algorithmes. Introduction à la logique et à la calculabilité*, Les Éditions de l'École polytechnique, 2010.

[26] G. DOWEK. *Proofs and Algorithms: An Introduction to Logic and Computability*, Springer, 2010, to be published.

[27] G. DOWEK, J.-J. LÉVY. *Introduction to the Theory of Programming Languages*, Springer, 2010.

[28] A. SPIWACK, T. COQUAND. *Constructively Finite?*, in "Contribuciones científicas en honor de Mirian Andrés Gómez", L. LAMBÁN PARDO, A. ROMERO IBÁÑEZ, J. RUBIO GARCÍA (editors), Universidad de La Rioja, 2010, p. 217-230, http://hal.inria.fr/inria-00503917/en/.

Other Publications

[29] P. ARRIGHI, G. DOWEK. *The physical Church thesis and the principles of quantum theory*, 2010, submitted to publication.

[30] G. DOWEK. *The physical Church thesis as an explanation of the Galileo thesis*, 2010, submitted to publication.

[31] G. DOWEK, M. J. GABBAY. *A translation of PNL to HOL*, 2010, submitted to publication.

[32] G. DOWEK, Y. JIANG. *On the expressive power of schemes*, 2010, to appear.

References in notes

[33] B. BARRAS, B. BERNARDO. *The Implicit Calculus of Constructions as a Programming Language with Dependent Types*, in "FoSSaCS", R. M. AMADIO (editor), Lecture Notes in Computer Science, Springer, 2008, vol. 4962, p. 365-379.