IN PARTNERSHIP WITH:
ENS Paris-Saclay

Activity Report 2018

Project-Team DEDUCTEAM

DEDUCTEAM

IN COLLABORATION WITH: Laboratoire specification et vérification (LSV)

RESEARCH CENTER
Saclay - Île-de-France

THEME
Proofs and Verification

Table of contents

1. Team, Visitors, External Collaborators **1**
2. Overall Objectives **2**
2.1. Objectives 2
2.2. History 2
3. Research Program **2**
3.1. Logical Frameworks 2
3.2. Interoperability and proof encyclopediae 3
3.3. Interactive theorem proving 3
4. Application Domains **3**
5. Highlights of the Year **4**
6. New Software and Platforms **4**
6.1. Autotheo 4
6.2. CoLoR 4
6.3. Coqine 4
6.4. Dedukti 5
6.5. Holide 5
6.6. HOT 5
6.7. iProver Modulo 5
6.8. mSAT 6
6.9. Rainbow 6
6.10. Krajono 6
6.11. archsat 7
6.12. lrat2dk 7
6.13. ekstrakto 7
6.14. SizeChangeTool 7
7. New Results **7**
7.1. $\lambda\Pi$-calculus modulo theory 7
7.2. Dedukti 8
7.3. Theories 8
7.4. Interoperability 9
7.5. Drags 9
7.6. SCTL 9
8. Partnerships and Cooperations **10**
8.1. National Initiatives 10
8.2. International Initiatives 10
8.3. Informal International Partners 10
8.4. International Research Visitors 10
9. Dissemination **10**
9.1. Promoting Scientific Activities 10
9.1.1. Scientific Events Organization 10
9.1.2. Scientific Events Selection 10
9.1.3. Journals 11
9.1.4. Invited Talks 11
9.1.5. Seminars 11
9.1.6. Leadership within the scientific community 11
9.1.7. Scientific Expertise 12
9.2. Teaching - Supervision - Juries 12
9.2.1. Teaching 12
9.2.2. Supervision 12

9.2.3. Juries 13
9.3. Popularization 13
9.3.1. Articles and contents 13
9.3.2. Education 13
9.3.3. Interventions 13
10. Bibliography **13**

Project-Team DEDUCTEAM

Creation of the Team: 2011 December 01, updated into Project-Team: 2017 January 01

Keywords:

Computer Science and Digital Science:

A2.1.4. - Functional programming
A2.1.11. - Proof languages
A2.4.3. - Proofs
A3.1.1. - Modeling, representation
A7. - Theory of computation
A7.2. - Logic in Computer Science

Other Research Topics and Application Domains:

B7. - Transport and logistics

1. Team, Visitors, External Collaborators

Research Scientists

Gilles Dowek [Team leader, Inria, Senior Researcher, HDR]
Bruno Barras [Inria, Researcher]
Frédéric Blanqui [Inria, Researcher, HDR]

Faculty Member

Guillaume Burel [ENSIIE, Associate professor, En délégation]

Post-Doctoral Fellows

Rodolphe Lepigre [Inria]
Franck Slama [Inria, from May 2018]

PhD Students

Guillaume Bury [Université Paris Diderot, until Sep 2018]
Mohamed Yacine El Haddad [Université de Paris-Saclay]
Gaspard Férey [Univserité de Paris-Saclay]
Guillaume Genestier [Université de Paris-Saclay]
François Thiré [Université de Paris-Saclay]

Interns

Ismail Lachheb [CNRS, from May 2018 until Jul 2018]
Walid Moustaoui [Inria, from May 2018 until Jul 2018]
Aristomenis Papadopoulos [Inria, from Jun 2018 until Sep 2018]
Quentin Ye [Inria, from May 2018 until Jul 2018]

Administrative Assistants

Adeline Lochet [Inria, from Jul 2018]
Emmanuelle Perrot [Inria, until Jun 2018]

External Collaborators

Jean-Pierre Jouannaud [Emeritus, HDR]
Catherine Dubois [ENSIIE, HDR]
Olivier Hermant [École Nationale Supérieure des Mines de Paris, HDR]

2. Overall Objectives

2.1. Objectives

The project-team investigates the design of logical frameworks, in order to ensure interoperability between proof systems, and to the development of system-independent proof libraries. To achieve these goals, we develop

- a logical framework DEDUKTI, where several theories can be expressed,
- tools to import proofs developed in external proof systems to DEDUKTI theories,
- tools to translate proofs from one DEDUKTI theory to another,
- tools to export proofs expressed in DEDUKTI theories to an external proof system,
- tools to prove the confluence, the termination, and the consistency of theories expressed in DEDUKTI,
- tools to develop proofs directly in DEDUKTI,
- an encyclopedia LOGIPEDIA of proofs expressed in various DEDUKTI theories.

2.2. History

The idea that systems such as Euclidean geometry or set theory should be expressed, not as independent systems, but in a logical framework appeared with the design of the first logical framework: predicate logic, in 1928. Later, several more powerful logical frameworks have been designed: λ-prolog, Isabelle, the Edinburgh logical framework, Pure type systems, and Deduction modulo theory.

The logical framework that we use is a simple λ-calculus with dependent types and rewrite rules, called the $\lambda\Pi$-calculus modulo theory, and also the Martin-Löf logical framework, and it generalizes all the mentioned frameworks. It is implemented in the system DEDUKTI.

The first version of DEDUKTI was developed in 2011 by Mathieu Boespflug [29]. From 2012 to 2015, new versions of DEDUKTI were developed and several theories were expressed in DEDUKTI, allowing to import proofs developed in MATITA (with the tool KRAJONO), HOL LIGHT (with the tool HOLIDE), FOCALIZE (with the tool FOCALIDE), IPROVER, and ZENON, totalizing several hundred of megabytes of proofs.

From 2015 to 2018, we focused on the translation of proofs from one DEDUKTI theory to another and to the exporting of proofs to other proof systems. In particular the MATITA arithmetic library has been translated to a much weaker theory: constructive simple type theory, allowing to export it to COQ, LEAN, PVS, HOL LIGHT, and ISABELLE/HOL. This led us to develop, in 2018, an online proof encyclopedia LOGIPEDIA, allowing to share and browse this library. We also focused on the development of new theories in DEDUKTI, and on an interactive theorem prover on top of DEDUKTI.

3. Research Program

3.1. Logical Frameworks

A thesis, which is at the root of our research effort, is that logical systems should be expressed as theories in a logical framework. As a consequence, proof-checking systems should not be focused on one theory, such as Simple type theory, Martin-Löf's type theory, or the Calculus of constructions, but should be theory independent. On the more theoretical side, the proof search algorithms, or the algorithmic interpretation of proofs should not depend on the theory in which proofs are expressed, but this theory should just be a parameter. This is for instance expressed in the title of our invited talk at ICALP 2012: *A theory independent Curry-De Bruijn-Howard correspondence* [31].

Various limits of Predicate logic have led to the development of various families of logical frameworks: λ-prolog and Isabelle have allowed terms containing free variables, the Edinburgh logical framework has allowed proofs to be expressed as λ-terms, Pure type systems have allowed propositions to be considered as terms, and Deduction modulo theory has allowed theories to be defined not only with axioms, but also with computation rules.

The $\lambda\Pi$-calculus modulo theory, that is implemented in the system DEDUKTI and that is a synthesis of the Edinburgh logical framework and of Deduction modulo theory, subsumes them all. Part of our research effort is focused on improving the $\lambda\Pi$-calculus modulo theory, for instance allowing to define congruences with associative and commutative rewriting. Another part of our research effort is focused on the automatic analysis of theories to prove their confluence, termination, and consistency either by pencil and paper proofs or automatically [4].

3.2. Interoperability and proof encyclopediae

Using a single prover to check proofs coming from different systems naturally leads to investigate how these proofs can be translated from one theory to another and used in a system different from the system in which they have been developed. This issue is of prime importance because developments in proof systems are getting bigger and, unlike other communities in computer science, the proof checking community has given little effort in the direction of standardization and interoperability.

For each proof, independently of the system in which it has been developed, we should be able to identify the systems in which it can be expressed. For instance, we have shown that many proofs developed in the MATITA prover did not use the full strength of the logic of MATITA and could be exported, for instance, to the systems of the HOL family, that are based on a weaker logic.

Rather than importing proofs from one system, transforming them, and exporting them to another system, we can use the same tools to develop system-independent proof encyclopedia. In such a library, each proof is labeled with the theories in which it can be expressed and so with the systems in which it can be used.

3.3. Interactive theorem proving

If our main goal with DEDUKTI is to import, transform, and export proofs developed in other systems, we also want to investigate how DEDUKTI can be used as the basis of an interactive theorem prover. This leads to two new scientific questions: first, how much can a tactic system be theory independent, and then how does rewriting extends the possibility to write tactics.

This has led to the development of a new version of DEDUKTI, which supports metavariables. Several tactics have been developed for this system, which are intended to help a human user to write proofs in our system instead of writing proof terms by hand. This work is a continuation of the previous work the team did on DEMON, which was an extension of DEDUKTI, whereas the support for interactive theorem proving is now native in DEDUKTI.

4. Application Domains

4.1. Interoperability

Our main impact applications, for instance to proofs of programs, or to air traffic control, are through our cooperation with other teams.

As a matter of fact, we view our work on interoperability and on the design of a formal proof encyclopedia as a service to the formal proof community.

5. Highlights of the Year

5.1. Highlights of the Year

Logipedia

We have launched in September the first system independent encyclopedia of formal proofs: LOGIPEDIA.

Awards

Serge Abiteboul and Gilles Dowek have received the Award *La science se livre* in January.

6. New Software and Platforms

6.1. Autotheo

KEYWORD: Automated deduction
SCIENTIFIC DESCRIPTION: Transformation of axiomatic theories into rewriting systems that can be used by iProverModulo.
FUNCTIONAL DESCRIPTION: Autotheo is a tool that transforms axiomatic theories into polarized rewriting systems, thus making them usable in iProverModulo. It supports several strategies to orient the axioms, some of them being proved to be complete, in the sense that ordered polarized resolution modulo the resulting systems is refutationally complete, some others being merely heuristics. In practice, Autotheo takes a TPTP input file and produces an input file for iProverModulo.
NEWS OF THE YEAR: Maintenance.

- Participant: Guillaume Burel
- Partner: ENSIIE
- Contact: Guillaume Burel
- Publication: Consistency Implies Cut Admissibility
- URL: http://www.ensiie.fr/~guillaume.burel/blackandwhite_autotheo.html.en

6.2. CoLoR

Coq Library on Rewriting and termination
KEYWORDS: Coq - Formalisation
FUNCTIONAL DESCRIPTION: CoLoR is a Coq library on rewriting theory and termination. It provides many definitions and theorems on various mathematical structures (quasi-ordered sets, relations, ordered semi-rings, etc.), data structures (lists, vectors, matrices, polynomials, finite graphs), term structures (strings, first-order terms, lambda-terms, etc.), transformation techniques (dependency pairs, semantic labeling, etc.) and (non-)termination criteria (polynomial and matrix interpretations, recursive path ordering, computability closure, etc.).

- Authors: Frédéric Blanqui and Sébastien Hinderer
- Contact: Frédéric Blanqui
- Publications: CoLoR: a Coq library on well-founded rewrite relations and its application to the automated verification of termination certificates - Automated Verification of Termination Certificates - CoLoR: a Coq library on rewriting and termination
- URL: http://color.inria.fr/

6.3. Coqine

Coq In dEdukti

KEYWORDS: Higher-order logic - Formal methods - Proof
FUNCTIONAL DESCRIPTION: CoqInE is a plugin for the Coq software translating Coq proofs into Dedukti terms. It provides a Dedukti signature file faithfully encoding the underlying theory of Coq (or a sufficiently large subset of it). Current development is mostly focused on implementing support for Coq universe polymorphism. The generated ouput is meant to be type-checkable using the latest version of Dedukti.

- Contact: Guillaume Burel
- URL: http://www.ensiie.fr/~guillaume.burel/blackandwhite_coqInE.html.en

6.4. Dedukti

KEYWORD: Logical Framework
FUNCTIONAL DESCRIPTION: Dedukti is a proof-checker for the LambdaPi-calculus modulo. As it can be parametrized by an arbitrary set of rewrite rules, defining an equivalence relation, this calculus can express many different theories. Dedukti has been created for this purpose: to allow the interoperability of different theories.

Dedukti's core is based on the standard algorithm for type-checking semi-full pure type systems and implements a state-of-the-art reduction machine inspired from Matita's and modified to deal with rewrite rules.

Dedukti's input language features term declarations and definitions (opaque or not) and rewrite rule definitions. A basic module system allows the user to organize his project in different files and compile them separately.

Dedukti features matching modulo beta for a large class of patterns called Miller's patterns, allowing for more rewriting rules to be implemented in Dedukti.
NEWS OF THE YEAR: There has been a new release 2.6 in 2018. This release provides a better control on module loading, and a better log of rewrite steps.

- Participants: François Thiré, Gaspard Ferey, Guillaume Genestier and Rodolphe Lepigre
- Contact: François Thiré
- Publications: Dedukti:un vérificateur de preuves universel - Rewriting Modulo β in the $\lambda\Pi$-Calculus Modulo - Expressing theories in the $\lambda\Pi$-calculus modulo theory and in the Dedukti system
- URL: https://deducteam.github.io/

6.5. Holide

KEYWORD: Proof
FUNCTIONAL DESCRIPTION: Holide translates HOL proofs to Dedukti[OT] proofs, using the OpenTheory standard (common to HOL Light and HOL4). Dedukti[OT] being the encoding of OpenTheory in Dedukti.

- Contact: Guillaume Burel
- URL: http://deducteam.gforge.inria.fr/holide/

6.6. HOT

Higher-Order Termination
FUNCTIONAL DESCRIPTION: HOT is an automated termination prover for higher-order rewriting, based on the notion of computability closure.

- Contact: Frédéric Blanqui
- URL: http://rewriting.gforge.inria.fr/hot.html

6.7. iProver Modulo

KEYWORDS: Automated deduction - Automated theorem proving
SCIENTIFIC DESCRIPTION: Integration of ordered polarized resolution modulo theory into the prover iProver.

FUNCTIONAL DESCRIPTION: iProver Modulo is an extension of the automated theorem prover iProver originally developed by Konstantin Korovin at the University of Manchester. It implements ordered polarized resolution modulo theory, a refinement of the resolution method based on deduction modulo theory. It takes as input a proposition in predicate logic and a clausal rewriting system defining the theory in which the formula has to be proved. Normalization with respect to the term rewriting rules is performed very efficiently through translation into OCaml code, compilation and dynamic linking. Experiments have shown that ordered polarized resolution modulo dramatically improves proof search compared to using raw axioms.
NEWS OF THE YEAR: Maintenance of Dedukti output

- Participant: Guillaume Burel
- Partner: ENSIIE
- Contact: Guillaume Burel
- Publications: A Shallow Embedding of Resolution and Superposition Proofs into the ??-Calculus Modulo - Experimenting with deduction modulo
- URL: https://github.com/gburel/iProverModulo

6.8. mSAT

KEYWORD: Propositional logic
FUNCTIONAL DESCRIPTION: mSAT is a modular, proof-producing, SAT and SMT core based on Alt-Ergo Zero, written in OCaml. The solver accepts user-defined terms, formulas and theory, making it a good tool for experimenting. This tool produces resolution proofs as trees in which the leaves are user-defined proof of lemmas.

- Contact: Guillaume Bury
- Publication: mSAT:An OCaml SAT Solver
- URL: https://github.com/Gbury/mSAT

6.9. Rainbow

Termination certificate verifier
KEYWORDS: Demonstration - Code generation - Verification
FUNCTIONAL DESCRIPTION: Rainbow is a set of tools for automatically verifying the correctness of termination certificates expressed in the CPF format used in the annual international competition of termination tools. It contains: a tool xsd2coq for generating Coq data types for representing XML files valid with respect to some XML Schema, a tool xsd2ml for generating OCaml data types and functions for parsing XML files valid with respect to some XML Schema, a tool for translating a CPF file into a Coq script, and a standalone Coq certified tool for verifying the correctness of a CPF file.

- Author: Frédéric Blanqui
- Contact: Frédéric Blanqui
- Publications: Automated verification of termination certificates - Automated verification of termination certificates
- URL: http://color.inria.fr/rainbow.html

6.10. Krajono

KEYWORD: Proof
FUNCTIONAL DESCRIPTION: Krajono translates Matita proofs into Dedukti[CiC] (encoding of CiC in Dedukti) terms.

- Contact: François Thiré

6.11. archsat

KEYWORDS: Automated theorem proving - First-order logic - Propositional logic
FUNCTIONAL DESCRIPTION: Archsat is an automated theorem prover aimed at studying the integration of first-order theorem prover technologies, such as rewriting, into SMT solvers.

- Contact: Guillaume Bury
- URL: https://gforge.inria.fr/projects/archsat

6.12. lrat2dk

KEYWORDS: Automated theorem proving - Proof
FUNCTIONAL DESCRIPTION: Take as input a SAT proof trace in LRAT format, which can be obtained from the de facto standard format DRAT using drat-trim. Output a proof checkable by Dedukti, in a shallow encoding of propositional logic.

- Participant: Guillaume Burel
- Partner: ENSIIE
- Contact: Guillaume Burel
- URL: https://github.com/gburel/lrat2dk

6.13. ekstrakto

KEYWORDS: TPTP - TSTP - Proof assistant - Dedukti
FUNCTIONAL DESCRIPTION: Extracting TPTP problems from a TSTP trace. Proof reconstruction in Dedukti from TSTP trace.

- Contact: Mohamed Yacine El Haddad
- URL: https://github.com/elhaddadyacine/ekstrakto

6.14. SizeChangeTool

KEYWORDS: Rewriting systems - Proof assistant - Termination
FUNCTIONAL DESCRIPTION: A termination-checker for higher-order rewriting with dependent types. Took part in the Termination Competition 2018 (http://termination-portal.org/wiki/Termination_Competition_2018) in the "Higher-Order Rewriting (union Beta)" category.

- Partner: Mines ParisTech
- Contact: Guillaume Genestier
- URL: https://github.com/Deducteam/SizeChangeTool

7. New Results

7.1. $\lambda\Pi$-calculus modulo theory

Gilles Dowek, Jean-Pierre Jouannaud and Jiaxiang Liu have started a program for developing new techniques for proving confluence of dependently typed theories, which do not rely on termination. These results have been presented at Types 2016, and will be submitted to a Journal early 2019. Target applications for these techniques are encodings of the Calculus of inductive constructions with polymorphic universes in the $\lambda\Pi$-calculus modulo theory.

Frédéric Blanqui has published in the Journal of Functional Programming a long article synthesizing his work on the use of size annotations for proving termination [12]. This paper provides a general and modular criterion for the termination of simply-typed λ-calculus extended with function symbols defined by user-defined rewrite rules. Following a work of Hughes, Pareto and Sabry, for functions defined with a fixpoint operator and pattern-matching, several criteria use typing rules for bounding the height of arguments in function calls. In this paper, we extend this approach to rewriting-based function definitions and more general user-defined notions of size.

Size-change termination is a technique introduced for first-order functional programs. In [16], Frédéric Blanqui and Guillaume Genestier show how it can be used to study the termination of higher-order rewriting in the $\lambda\Pi$-calculus modulo theory.

Dependency pairs are a key concept at the core of modern automated termination provers for first-order term rewrite systems. In [22], Frédéric Blanqui, Guillaume Genestier and Olivier Hermant introduced an extension of this technique for a large class of dependently-typed higher-order rewrite systems. This improves previous results by Wahlstedt on one hand and Frédéric Blanqui on the other hand to strong normalization and non-orthogonal rewrite systems. This new criterion has been implemented in the type-checker DEDUKTI.

7.2. Dedukti

Frédéric Blanqui and Guillaume Genestier have formally defined the operational semantics of DEDUKTI 2.5, showing some problems with non left-linear rewrite rules.

Rodolphe Lepigre, Frédéric Blanqui and Franck Slama developed a new version of DEDUKTI, available on https://github.com/Deducteam/lambdapi, with meta-variables and a small set of tactics in order to be able to build DEDUKTI proofs interactively.

Aristomenis-Dionysios Papadopoulos has added a rewrite tactic in the style of Ssreflect [27].

Emilio Gallego added an LSP server for communicating with editors.

Ismail Lachheb has developed a plugin for DEDUKTI based on the LSP protocol into the Atom editor [25].

Guillaume Burel added support for polarized Deduction modulo theory in DEDUKTI.

Quentin Ye has developed an algorithm to compare λ-terms. The main point was to take sharing into account, so as to relate the complexity with the space used to represent the term, rather than with the size of the term. He has implemented this algorithm in the DEDUKTI codebase. He has also run his algorithm on examples that show an exponential speed-up compared to the naive algorithm [21].

7.3. Theories

Gaspard Férey and François Thiré defined a new encoding for Cumulative type systems (CTS) in the $\lambda\Pi$-calculus modulo theory, extending the work of Ali Assaf's PhD [28]. This encoding relies on explicit subtyping which requires additional computational rules. It provides a way to encode a larger class of CTS, which sheds a new light on the computational content of explicit subtyping. This encoding should be extendable to express more advanced features such as universe polymorphism in the Calculus of Inductive Construction, a first step to have a faithful encoding of the COQ system. The encoding has been proven correct under the hypothesis that the computational rules are confluent.

François Thiré redesigned the tool UNIVERSO, so that it can be used for a larger class of CTS. The specification for UNIVERSO can be given by rewrite rules which makes UNIVERSO much easier to use. This tool is a first step to have an automatic chain of translations to translate proofs in the encoding of MATITA to $\mathrm{STT}\forall_{\beta\delta}$.which would make these proofs interoperable with 5 different systems.

François Thiré changed the encoding provided by KRAJONO to integrate some ideas of the encoding discussed above. This encoding is compatible with the tool UNIVERSO.

Gaspard Férey updated the COQINE software to translate COQ's 8.8 version. In this version, the standard library relies on universe polymorphism so partial support for the translation of this feature was integrated. Since encodings of the many features of Coq (inductive constructions, floating universes, several kinds of universe polymorphisms, etc) are a current work in progress, the software was made parameterizable to allow experimentations of multiple encodings of these features.

Gaspard Férey showcased an encoding of the Calculus of Inductive Constructions (CiC) relying on associative-commutative (AC) rewriting on the arithmetic library translated from MATITA. This practical experiment shows the limitations of AC-rewriting (as implemented in DEDUKTI) in terms of performance and the need for special care when defining encodings relying on this feature.

Guillaume Burel began to write a tool translating SAT proof traces in LRAT format into DEDUKTI proofs. The main issue was that steps in LRAT traces are not logical consequences of previous clauses but only preserve provability.

Mohamed Yacine El Haddad developed a tool to extract TPTP problems from a TSTP trace (generated by automated theorem provers) and reconstruct the proof of the trace in DEDUKTI format.

Bruno Barras has started to develop a model of Homotopy Type Theory (HoTT) in DEDUKTI. This is basically a presheaf model, where the choice of the base category leads either to the simplicial sets model or to the cubical model of HoTT. This construction generalizes the setoid model construction [2] to an arbitrary dimension. Since this involves encoding notions of category theory, the rewriting feature of DEDUKTI is intensively used to represent, among others, the associativity of morphism composition, or the naturality conditions.

Guillaume Bury has proposed an automation-friendly set theory for the B method. This theory is expressed using first order logic extended to polymorphic types and rewriting. Rewriting is introduced along the lines of deduction modulo theory, where axioms are turned into rewrite rules over both propositions and terms. This work has been published in [30].

7.4. Interoperability

François Thiré has defined in DEDUKTI a constructive version of simple type theory with prenex polymorphism: STT$\forall_{\beta\delta}$. This work has been published at the LFMTP workshop in [15]. STT$\forall_{\beta\delta}$ has been used to encode an arithmetic library able to prove little Fermat's theorem. Then these proofs has been exported to different systems that are: COQ, MATITA, LEAN and OPENTHEORY. Gilles Dowek, César Muñoz, and François Thiré have developed a translation of STT$\forall_{\beta\delta}$ to PVS.

Then, Walid Moustaoui and François Thiré have built a website called LOGIPEDIA which allows the user to inspect this arithmetic library and the user can download the proof of this theorem to one of the systems mentioned above.

7.5. Drags

Shared and cyclic structures are very common in both programming and proving, which requires generalizing term rewriting techniques to graphs. Jean-Pierre Jouannaud and Nachum Dershowitz have introduced a very general class of multigraphs, called drags, equipped with a composition operator $\otimes$ which provides with a rich categorical structure. Rewriting a drag D can then be defined in a very simple way, by writing D as the composition of a left-hand side of rules L and a context C, and then replacing L by R, the right-hand side of the rule, which yields the rewritten drag $R \otimes C$. The fundamental aspects of the algebra of drags have been presented at TERMGRAPH'2018 and have also been submitted to a special issue of TCS. Termination of drag rewriting in investigated in [20].

7.6. SCTL

Gilles Dowek, Liu Jian, and Ying Jiang have reworked the presentation of CTL in sequent calculus proposed by Gilles Dowek and Ying Jiang in 2012 and provided an implementation of it. This work has been published in [13].

8. Partnerships and Cooperations

8.1. National Initiatives

The ANR PROGRAMme is an ANR for junior researcher Liesbeth Demol (CNRS, UMR 8163 STL, University Lille 3) to which G. Dowek participates. The subject is: "What is a program? Historical and Philosophical perspectives". This project aims at developing the first coherent analysis and pluralistic understanding of "program" and its implications to theory and practice.

8.2. International Initiatives

Brazil: STIC Amsud.

Argentina: Ecos

China: Inria-NSFC

8.3. Informal International Partners

Our main international partners are Alejandro Diáz-Caro (Buenos Aires), Bruno Lopes (Niteroi), Ying Jiang (Beijing), Florian Rabe (Bremen), Brigitte Pientka (McGill), César Muñoz (NASA), and Stéphane Graham-Lengrand (SRI).

8.4. International Research Visitors

Alejandro Díaz-Caro (Buenos Aires) has visited Deducteam for two weeks.

Ying Jiang (Beijing) has visited Deducteam for three weeks.

Aristomenis-Dionysios Papadopoulos (Imperial College, London) has visited Deducteam. He worked with Frédéric Blanqui on the development of a rewrite tactic in DEDUKTI [27].

8.4.1. Visits to International Teams

Gilles Dowek has spent two weeks at the University of Buenos Aires.

Gilles Dowek has spent two weeks at the Institute of Aerospace (USA).

9. Dissemination

9.1. Promoting Scientific Activities

9.1.1. Scientific Events Organization

Guillaume Burel has been local organizer of the scientific days of the CNRS GDR GPL working groups LTP and MTV2.

9.1.2. Scientific Events Selection

Frédéric Blanqui has been PC chair of the 13th International Workshop on Logical Frameworks and Meta-Languages: Theory and Practice (LFMTP'18) with Giselle Reis.

Frédéric Blanqui is Workshop Chair of LICS and member of the Steering Committee of LICS.

Frédéric Blanqui is member of the Steering Committee of the International School on Rewriting (ISR) of the WG 1.6 of the International Federation for Information Processing.

Gilles Dowek has been a PC member of TYPES 2018.

Guillaume Burel has been PC member of the 30th Journées Francophones des Langages Applicatifs.

Guillaume Burel has reviewed a submission for the International Conference on Principles and Practice of Constraint Programming (CP). Guillaume Genestier reviewed submissions to the conferences Logic in Computer Science (LICS), Principles and Practice of Declarative Programming (PPDP) and European Symposium on Programming (ESOP).

9.1.3. Journals

Gilles Dowek is an editor of TCS-C.

Frédéric Blanqui has reviewed a paper for Mathematical Structures in Computer Science (MSCS). Guillaume Burel has reviewed papers for the Computer Journal and Logical Methods in Computer Science (LMCS). Rodolphe Lepigre has reviewed a paper for International Conference on Foundations of Software Science and Computation Structures (FoSSaCS). Rodolphe Lepigre has reviewed a paper for the journal ACM Transactions on Programming Languages and Systems (TOPLAS). Franck Slama has reviewed a paper for the Journal of Functional Programming.

9.1.4. Invited Talks

- Rodolphe Lepigre gave an invited talk entitled “The PML Language: Realizability at the Service of Program Proofs” at the Realizability Workshop (12-13 June 2018) in Luminy.
- Rodolphe Lepigre gave an invited talk entitled “An Overview of the PML$_2$ Language: Realizability, Subtyping and Cyclic Proofs” at LRI, for the starting days of the new Scalp working group of GDR IM. This is a presentation of his paper [32].
- Gilles Dowek has given an invited talk at NFM (Nasa Formal Methods).
- Jean-Pierre Jouannaud has given an invited talk at the workshop "Rewriting Techniques for Program Transformation and Evaluation" at FLoC, on July 8, 2018.

9.1.5. Seminars

- Gilles Dowek has participated to the meeting "From Information to Cells" organized by Hélène Kirchner and Antoine Danchin. He has given a talk at the National Institute of Aerospace.
- Gilles Dowek has co-organized a seminar on Logic and Philosophy at the CNFHPST.
- Guillaume Burel has presented a talk entitled “Bridging holes on DEDUKTI proofs, an overview” at the scientific day of the Digicosme working group UPSCaLe.
- Bruno Barras has given a talk entitled “An analysis of bindlib” at the UPSCaLe meeting (June’18) held in Palaiseau.
- Mohamed Yacine EL HADDAD has presented his work at internal laboratory seminar of LSV (June’18) and SAMOVAR (November’18).
- Gaspard Férey has presented his work at internal laboratory seminar of LSV (June’18).
- Guillaume Genestier has presented his work at the internal laboratory seminar of Centre de Recherche en Informatique of Mines ParisTech (February’18) and LSV (June’18) and presented DEDUKTI at the doctoral seminar of La Société Informatique de France (June’18). He presented [16] in the WorkShop on Termination (WST) at Oxford (July’18).
- Rodolphe Lepigre has presented his work on "Termination checking using well-founded typing derivations" at a Deducteam seminar in September 2018.
- Rodolphe Lepigre has given a talk entitled “The PML$_2$ Language, Integrated Program Verification in ML” at the Max Planck Institute for Software Systems in Saarbrücken, in November 2018.
- Franck Slama has presented some previous work at an internal laboratory seminar of LSV in December 2017.
- François Thiré has presented his work on interoperability at the UPSCaLe seminar on March 2018, then he presented his paper [15] at the LFMT Workshop at Oxford (July’17).
- Aristomenis Papadopoulos has presented the work he did during his summer internship at a Deducteam seminar in September 2018.

9.1.6. Leadership within the scientific community

Gilles Dowek is president of the scientific board of the Socitété informatique de France.

He is a member of the Ethic council CERNA.

He is a member of the Comité National Français d'Histoire et de Philosophie des Sciences et des techniques.

He is a member of the scientific board of La Main à la pâte.

He is a member of the scientific board of the Institut Villebon Charpak.

He is a member of the scientific board of the Maison des sciences de Lorraine.

He is the president of the Board of teacher school (ESPE) of the University of Lorraine.

He is a member of the scientific board of SystemX.

He is a member of the scientific board of the team Humanités numériques at the Collège des Bernardins.

Gilles Dowek and Jean-Pierre Jouannaud are honorary members of IFIP-WG1.6.

Jean-Pierre Jouannaud is a permanent member of the visiting committee of Academia Sinica, Taiwan.

9.1.7. Scientific Expertise

Frédéric Blanqui reviewed a project for the Netherlands Organization for Scientific Research (NWO).

9.2. Teaching - Supervision - Juries

9.2.1. Teaching

- Master: Bruno Barras, proof assistants, 12h, M2, MPRI
- Master: Frédéric Blanqui, formal languages, 21h, M1, ENSIIE
- Master: Frédéric Blanqui, rewriting theory, 14h, M1, ENS Paris-Saclay
- Master: Frédéric Blanqui, λ-calculus and theories in first-order logic, 18h, M1/M2, ENS Paris-Saclay
- Master: Gilles Dowek has given a course at MPRI.
- Master: Gilles Dowek is in charge of the second year of Masters at the École normale de Paris Saclay.
- Master: Gilles Dowek has given a one week invited course at the University of Buenos Aires.
- Licence: Guillaume Genestier, logic tutorials, 45h, L3, ENS Paris-Saclay
- Licence: Guillaume Genestier, complexity remedial classes, 11h, L3, ENS Paris-Saclay
- Licence: Gaspard Férey, language theory, 44h, L3, EISTI
- Licence: François Thiré, (spring) logic project, 26h, L3 ENS Paris-Saclay
- Licence: François Thiré, (spring) Programmation 2 tutorials, 26h, L3 ENS Paris-Saclay
- Licence: François Thiré, (fall) Architectures and Systems tutorials, 36h, L3 ENS Paris-Saclay
- Frédéric Blanqui is co-director of the pole 4 of the doctoral school STIC of the University Paris-Saclay.
- Frédéric Blanqui is member of the committee of the doctoral school of the ENS Paris-Saclay.
- Frédéric Blanqui is in charge of following PhD students at LSV.

9.2.2. Supervision

- PhD Defended: Frédéric Gilbert, Gilles Dowek and Florent Kirchner,
- PhD in progress: Guillaume Bury, David Delahaye and Gilles Dowek,
- PhD in progress: Guillaume Genestier, termination in $\lambda\Pi$-calculus modulo theory, 01/10/17, Frédéric Blanqui and Olivier Hermant,
- PhD in progress: Mohamed Yacine El Haddad, using automated provers in proof assistants, 05/01/18, Frédéric Blanqui and Guillaume Burel,

- PhD in progress: Gaspard Férey, Associative-Commutative rewriting in the $\lambda\Pi$-calculus, 01/09/18, Gilles Dowek,
- PhD in progress: François Thiré, Design tools to make interoperability easier in DEDUKTI, 01/09/18, Gilles Dowek.

9.2.3. Juries

Gilles Dowek has been a member of the Jury of the PhD defence of Pierre Boutry. He has been an evaluator of the thesis of Thibault Gauthier. He has been a member of the Jury of the habilitation defence of Julien Signoles and of Alexei Grinbaum.

9.3. Popularization

9.3.1. Articles and contents

Gilles Dowek writes a monthly column in Pour la Science (12 issues) and has started a bi-monthly column in Le Monde (3 issues).

Gilles Dowek has given interviews to France Inter, Radio France Internationale, France Culture, Ouest France, Usbek et Rica, and Philosophie Magazine.

9.3.2. Education

Gilles Dowek has participated to meetings on scientific education in Switzerland, Belgium, and Côte d'Ivoire.

He has been heard by a committee of the the Éducation Nationale on pedagogical data and privacy.

He has given a talk on job mutations to mathematics inspectors.

9.3.3. Interventions

Gilles Dowek has given popular science talks in Toulouse, Antony, Issoudun, Rueil Malmaison, Saint Louis, Saint-Cloud, Rennes, Nancy, Paris, Nîmes, St Quentin en Yvelines, Montbéliard, Molaix, St Agrève, Rhodes, Marcoule, and Juvisy.

10. Bibliography

Major publications by the team in recent years

[1] A. Assaf, G. Burel, R. Cauderlier, D. Delahaye, G. Dowek, C. Dubois, F. Gilbert, P. Halmagrand, O. Hermant, R. Saillard. *Expressing theories in the $\lambda\Pi$-calculus modulo theory and in the Dedukti system*, in "22nd International Conference on Types for Proofs and Programs, TYPES 2016", Novi SAd, Serbia, May 2016, https://hal-mines-paristech.archives-ouvertes.fr/hal-01441751

[2] B. Barras, T. Coquand, S. Huber. *A generalization of the Takeuti-Gandy interpretation*, in "Mathematical Structures in Computer Science", 2015, vol. 25, n^{o} 5, pp. 1071–1099, https://doi.org/10.1017/S0960129514000504

[3] F. Blanqui. *Definitions by rewriting in the Calculus of Constructions*, in "Mathematical Structures in Computer Science", 2005, vol. 15, n^{o} 1, pp. 37-92 [*DOI :* 10.1017/S0960129504004426], http://hal.inria.fr/inria-00105648/en/

[4] F. Blanqui, J.-P. Jouannaud, A. Rubio. *The Computability Path Ordering*, in "Logical Methods in Computer Science", October 2015 [*DOI :* 10.2168/LMCS-11(4:3)2015], https://hal.inria.fr/hal-01163091

[5] G. Burel. *Experimenting with Deduction Modulo*, in "CADE 2011", V. Sofronie-Stokkermans, N. Bjørner (editors), Lecture Notes in Artificial Intelligence, Springer, 2011, vol. 6803, pp. 162–176

[6] D. Cousineau, G. Dowek. *Embedding Pure Type Systems in the $\lambda\Pi$-calculus modulo*, in "Typed lambda calculi and applications", S. Ronchi della Rocca (editor), Lecture Notes in Computer Science, Springer-Verlag, 2007, vol. 4583, pp. 102-117

[7] G. Dowek, T. Hardin, C. Kirchner. *Theorem proving modulo*, in "Journal of Automated Reasoning", 2003, vol. 31, pp. 33-73

[8] O. Hermant. *Resolution is Cut-Free*, in "Journal of Automated Reasoning", March 2010, vol. 44, n^{o} 3, pp. 245-276

[9] M. Jacquel, K. Berkani, D. Delahaye, C. Dubois. *Verifying B Proof Rules using Deep Embedding and Automated Theorem Proving*, in "Software and Systems Modeling (SoSyM)", June 2013

[10] M. Jacquel, K. Berkani, D. Delahaye, C. Dubois. *Tableaux Modulo Theories Using Superdeduction*, in "Global Journal of Advanced Software Engineering (GJASE)", December 2014, vol. 1, pp. 1 - 13 [*DOI :* 10.1007/978-3-642-31365-3_26], https://hal.archives-ouvertes.fr/hal-01099338

Publications of the year

Doctoral Dissertations and Habilitation Theses

[11] F. Gilbert. *Extending higher-order logic with predicate subtyping: Application to PVS*, Université Sorbonne Paris Cité ; Université Paris Diderot, April 2018, https://hal.inria.fr/hal-01673518

Articles in International Peer-Reviewed Journals

[12] F. Blanqui. *Size-based termination of higher-order rewriting*, in "Journal of Functional Programming", April 2018, https://arxiv.org/abs/1802.06603 [*DOI :* 10.1017/S0956796818000072], https://hal.inria.fr/hal-01424921

[13] Y. Jiang, J. Liu, G. Dowek, K. Ji. *Towards Combining Model Checking and Proof Checking*, in "The Computer Journal", 2019, https://hal.inria.fr/hal-01970274

[14] R. Lepigre, C. Raffalli. *Abstract Representation of Binders in OCaml using the Bindlib Library*, in "Electronic Proceedings in Theoretical Computer Science", July 2018, vol. 274, pp. 42-56, https://arxiv.org/abs/1807.01872 - In Proceedings LFMTP 2018, arXiv:1807.01352 [*DOI :* 10.4204/EPTCS.274.4], https://hal.inria.fr/hal-01972050

[15] F. Thiré. *Sharing a Library between Proof Assistants: Reaching out to the HOL Family* *, in "Electronic Proceedings in Theoretical Computer Science", July 2018, vol. 274, pp. 57 - 71 [*DOI :* 10.4204/EPTCS.274.5], https://hal.inria.fr/hal-01929714

International Conferences with Proceedings

[16] F. Blanqui, G. Genestier. *Termination of $\lambda\Pi$ modulo rewriting using the size-change principle (work in progress)*, in "16th International Workshop on Termination", Oxford, United Kingdom, S. Lucas (editor), July 2018, pp. 10-14, https://arxiv.org/abs/1812.01853 , https://hal.inria.fr/hal-01944731

[17] G. BUREL. *Linking Focusing and Resolution with Selection*, in "43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)", Liverpool, United Kingdom, I. POTAPOV, P. SPIRAKIS, J. WORRELL (editors), Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, August 2018, vol. 117, pp. 9:1-9:14 [*DOI :* 10.4230/LIPICS.MFCS.2018.9], https://hal.inria.fr/hal-01670476

[18] N. DERSHOWITZ, J.-P. JOUANNAUD. *Drags: A Simple Algebraic Framework For Graph Rewriting*, in "TERMGRAPH 2018 - 10th International Workshop on Computing with Terms and Graphs", Oxford, United Kingdom, July 2018, https://hal.inria.fr/hal-01853836

National Conferences with Proceedings

[19] S. COLIN, R. LEPIGRE, G. SCHERER. *Unboxing Mutually Recursive Type Definitions in OCaml*, in "JFLA 2019", Les Rousses, France, January 2019, https://arxiv.org/abs/1811.02300 , https://hal.inria.fr/hal-01929508

Conferences without Proceedings

[20] N. DERSHOWITZ, J.-P. JOUANNAUD. *Graph Path Orderings*, in "2nd International Conference on Logic for Programming, Artificial Intelligence and Reasoning", Awassa, Ethiopia, G. BARTHE, G. SUTCLIFFE, M. VEANES (editors), EPiC Series in Computing, November 2018, vol. 57, pp. 1-18, https://hal.inria.fr/hal-01903086

Research Reports

[21] Q. YE. *Comparaison des termes avec partage*, LSV, ENS Cachan, CNRS, Inria, Université Paris-Saclay, Cachan (France), July 2018, https://hal.inria.fr/hal-01973539

Other Publications

[22] F. BLANQUI, G. GENESTIER, O. HERMANT. *Dependency Pairs Termination in Dependent Type Theory Modulo Rewriting*, December 2018, working paper or preprint, https://hal.inria.fr/hal-01943941

[23] A. DÍAZ-CARO, G. DOWEK. *A logic identifying isomorphic propositions*, August 2018, https://arxiv.org/abs/1501.06125 - There is a mistake in the main proof (strong normalisation), and the system is actually not normalising in its current form, https://hal.inria.fr/hal-01109104

[24] F. GILBERT. *Verifiable certificates for predicate subtyping*, January 2019, working paper or preprint, https://hal.inria.fr/hal-01977585

[25] I. LACHHEB. *Une interface pour Dedukti*, LSV, ENS Cachan, CNRS, Inria, Université Paris-Saclay, Cachan (France), September 2018, https://hal.inria.fr/hal-01898401

[26] W. MOUSTAOUI. *Encyclopédie en ligne de démonstrations formelles*, LSV, ENS Cachan, CNRS, Inria, Université Paris-Saclay, Cachan (France), September 2018, https://hal.inria.fr/hal-01975446

[27] A.-D. PAPADOPOULOS. *Industrial Placement Report Research Placement in Deducteam*, Imperial College London, October 2018, https://hal.inria.fr/hal-01890253

References in notes

[28] A. ASSAF. *A framework for defining computational higher-order logics*, École polytechnique, September 2015, https://pastel.archives-ouvertes.fr/tel-01235303

[29] M. BOESPFLUG. *Conception d'un noyau de vérification de preuves pour le $\lambda\Pi$-calcul modulo*, École Polytechnique, 2011

[30] G. BURY, S. CRUANES, D. DELAHAYE, P. EUVRARD. *An Automation-Friendly Set Theory for the B Method*, in "Abstract State Machines, Alloy, B, TLA, VDM, and Z - 6th International Conference, ABZ 2018, Southampton, UK, June 5-8, 2018, Proceedings", M. J. BUTLER, A. RASCHKE, T. S. HOANG, K. REICHL (editors), Lecture Notes in Computer Science, Springer, 2018, vol. 10817, pp. 409–414, https://doi.org/10.1007/978-3-319-91271-4_32

[31] G. DOWEK. *A Theory Independent Curry-de Bruijn-howard Correspondence*, in "Proceedings of the 39th International Colloquium Conference on Automata, Languages, and Programming - Volume Part II", Berlin, Heidelberg, ICALP'12, Springer-Verlag, 2012, pp. 13–15, http://dx.doi.org/10.1007/978-3-642-31585-5_2

[32] R. LEPIGRE. *PML 2 : Integrated Program Verification in ML*, in "23rd International Conference on Types for Proofs and Programs (TYPES 2017)", Budapest, Hungary, July 2017, 27 p. [*DOI :* 10.4230/LIPICS.TYPES.2017.5], https://hal.inria.fr/hal-01972000