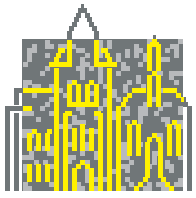
T. Coquand (Chalmers UT, Göteborg, S), H. Lombardi (Univ. de Franche Comté , F), M.-F. Roy (IRMAR, Rennes, F)
(Editors)

Verification and Constructive Algebra

Dagstuhl Seminar 03021 – January 05 to January 10, 2003
Dagstuhl-Seminar-Report No. 362



SCHLOSS DAGSTUHL

INTERNATIONALES
BEGEGNUNGS-
UND FORSCHUNGSZENTRUM
FÜR INFORMATIK

ISSN 0940-1121

Herausgegeben von IBFI gem. GmbH, Schloss Dagstuhl, 66687 Wadern, Germany.

Das Internationale Begegnungs- und Forschungszentrum für Informatik (IBFI) Schloss Dagstuhl ist eine gemeinnützige GmbH. Sie veranstaltet regelmäßig wissenschaftliche Seminare, welche nach Antrag der Tagungsleiter und Begutachtung durch das wissenschaftliche Direktorium mit persönlich eingeladenen Gästen durchgeführt werden.

Gesellschafter:

- Gesellschaft für Informatik e.V. – Bonn
- TH Darmstadt
- Universität Frankfurt
- Universität Kaiserslautern
- Universität Karlsruhe
- Universität Stuttgart
- Universität Trier
- Universität des Saarlandes

Verification and Constructive Algebra

Thierry Coquand Henri Lombardi Marie-Françoise Roy

5-10 january, 2003

General Presentation

The meeting was an attempt to bring together people from different communities: constructive algebra, computer algebra, designers and users of proof systems. Though the goals and interests are distinct, the meeting revealed that there is a strong core of common interests, the main one maybe the shared desire to understand in depth mathematics concepts in connections with algorithms and proofs. An interaction appears thus to be possible and fruitful. One outcome of this week was the decision to create a European group under the acronym MAP for "Mathematics: Algorithms and Proofs". As we said in our proposal: "If there is enough common interests and good interactions during the week, the Dagstuhl seminar could be the starting point of a european proposal on the same topic, with more ambitious goals." This is indeed what happened.

We would like at this point to thank the team of Schloss Dagtuhl. The exceptional working conditions we enjoyed there played an important rôle in the success of the meeting.

Summary of the meeting

Here are some common themes that emerged in the meeting on constructive algebra and verifications. There is no attempt to be exhaustive.

Certificates

A first common theme that emerged can be captured by the notion of "certificate", and was exposed clearly by the talk of Arjeh Cohen. This notion unifies some attempts to connect proof systems and computer algebra systems, that were the topic of the talks of Loic Pottier and David Delaye. The idea is roughly that computer algebra should communicate mathematical data together with a *certificate*, which represents the information needed to complete a proof of correctness of the mathematical data. This notion is reminiscent of the difference NP/P: it may be hard to check that a formula is a tautology but it is easy to check a proof. A simple example is provided by the gcd of two polynomials P and Q. The computer system should communicate not only the answer G, but also a certificate, that may be four polynomials A, B, C, D such that $AP + BQ = G, P = CG, Q = DG$. To find G may be hard, but to check these equalities is easy. A more sophiscated example was the topic of the talk of Loic Pottier (special cases of quantifier eliminations for reals), who had to program in CAML his own version of a computer algebra algorithm in order to get the desired certificates.

This notion of certificate is also closely connected to the talk of Helmut Schwichtenberg (common to all interactive proof systems with explicit proof objects): a starting point of such work is that while it is undecidable in general whether a given program meets its specification. In contrast, it can be checked easily by a machine whether a formal proof is correct. The proof object itself can thus then be used as a certificate.

It is curious that a similar notion of certificate was used in the talk of Dmitrii Pasechnik. There, of course, the goal is completely different, which is to provide interesting strong propositional proof systems with lower bound results. Finally, the talk of Laureano Gonzalez-Vega was concerned on the difficulty of computing algebraic certificates in some geometrical statements in Real Algebraic Geometry.

Algorithms in Mathematics, via Proof Theory

A second theme is what one may call the relevance of classical mathematics to algorithms. Tge talks of Henri Lombardi, Marie-Francoise Roy and Ulrich Kohlenbach showed, in very different ways, that mathematical proofs that use a priori highly non computational concepts, such as Zorn lemma, or compactness principles, contain implicitely very interesting computational informations. The talk of Ulrich Kohlenbach presented a way to extract implicit informations in proofs, in such a way that one can even obtain new theorems, surprising to the expert, from these informations (here in the field of metric fixed point theory). One interesting topic is to compare the two approaches: in Lombardi and Roy's talks, to use techniques from geometric logic, and in Kohlenbach's talk, a modification of Gödel's dialectica interpretation, that is especially well suited to extract bounds from classical proofs. Ulrich Kohlenbach said for instance that it should be interesting to use his methods also for examples on algebra, where the dynamical method of Lombardi-Roy has been used so far. A general feeling, emerging from some talks and discussions, was that the algorithms extracted by the dynamical method from a priori non effective proofs, may give algorithms that are better (even feasible) than the algorithms one can extract more straightforwardly from usual constructive arguments. For instance, in usual constructive mathematics, one requires to have a test of irreducibility for polynomials. While such a test exists in some cases, there are usually quite inefficient. The algorithm corresponding to a proof using this test is thus a priori also inefficient. By contrast the algorithm extracted from dynamical methods does not rely on such tests. It was suggested by Henri Lombardi that some efficient algorithms may be obtained in this way in number theory (dynamical theory of Dedekind domains). Such claims, if they happen to be verified, are of fundamental importance.

Progress on basics

Another theme is best expressed by one sentence taken from the presentation of the seminar: "It is remarkable that in constructive and computer algebra, progress in sophisticated algorithms often implies progress on basics". This point was stressed in the talk of Peter Paule on symbolic summation for instance, who provided basic examples that would be welcome additions to basic courses on calculus, and several time in discussions, for instance for algebraic topology. Another example was provided by the talk of Gilles Dowek, who, motivated by a quite concrete problems in safety of air trafic control, presented a new form of induction over real numbers that may be interesting for presenting basic proofs in real analysis.

Proof Systems and Computer Algebra Systems

A large part of the talks were concerned about connections between computer algebra systems and proof systems. Peter Paule reminded us, with some concrete examples, that people in proof system should be more aware of the power of current computer algebra systems. The talks of Renaud Rioboo presented a system aiming at combining proofs and computer algebra computations. The talks of Clemens Ballarin and Julio Rubio supplemented the talk of Francis Sergeraert by presenting an on-going attempt to use techniques from formal methods and interactive proof checking to ensure the correctness of a large sofware system for computations in algebraic topology. One interesting conceptual connection emerged from the talk of Peter Paule, on the concrete example of checking tables of equalities between special functions, like for instance $cos^2\ x + sin^2\ x = 1$. (There is actually a NIST-DLMF project of creating a new

Digital Library of Mathematica Functions, and verification is an important aspect of' this project.) What is done in computer algebra is a purely algebraic model of the problem (here for instance using differential algebra to show that the derivative of $cos^2\ x + sin^2\ x$ is 0.) But there is a mismatch between this representation and the representation of expressions as *functions* of real or complex quantities. Typically, the functions may have pôle, or may involve ambiguities. What interest primarily the user of such tables is of course the interpretation of expressions as functions. This suggests a natural place where proof systems may complement computer algebra system. Such a connection appeared in the talks of Loic Pottier and David Delaye. The simplest example may be the provided by the equality $x \times 1/x = 1$. This equality is perfectly valid from the computer algebra viewpoint, since it is interpreted in the field of rational expressions (field of fractions of a polynomial ring). Considered as a function $x \longmapsto 1/x$ has a pôle at $x = 0$ and the proof system will have to generate the condition $x \neq 0$.

Constructive Mathematics

Several talks were given on constructive mathematics. Francis Sergereart presented a way to do algebraic topology constructively, which is actually implemented in common lisp. Peter Schuster presented a constructive definition of the notion of scheme, a basic concept in modern algebraic geometry. There are probably deep connections between this presentation, based on point-free topology, and the talks of Henri Lombardi and Herve Perdry on dynamical algebras, that would be interesting to explore further. The talks of Erik Palmgren and Jesper Carlström were about Martin-Löf type theory. Type theory appears to be a potential formalism in which several concepts that were presented at the workshop could be elegantly expressed. Just to take one example, if we succeed to express constructive algebraic topology, as presented by Francis Sergeraert, in type theory, one would have an algorithm (in a functional programming language) which is correct by construction, thus bypassing the need of a formal verification a posteriori. In the present stage however, this may seem utopic (probably the program obtained in this way would be too inefficient), but this might be an interesting project. The meeting ended by a talk of Bas Spitters on a constructive proof of Peter-Weyl's theorem, and it would be interesting to explore further the algorithmic ideas implicit in this proof.

Impact

The main positive surprise of the seminar was that communication is possible, and in fact highly appreciated, bewteen quite distinct fields of mathematics and computer science. One participant expressed for instance his positive surprise to see in the same talk the name of Jean-Pierre Serre, who made fundamental contributions in algebraic topology, and the name of Turing, one of the founder of the mathematical notion of algorithm. The participants were working in different fields, but were all deeply interested in the interconnections between mathematics, algorithms and proofs, and several participants expressed the opinion that this combination of different topics with a strong common interest allows for a rich interaction. What was positive also was the emphasis, common to many talks, that progress in sophisticated mathematics and algorithms often implies progress on basics. This seminar was also a wellcome occasion to have a beginning of a real dialogue between designers and users of proof systems, and specialists in computer algebra and mathematics. Such dialogues have already started in research groups that were represented (Linz, Nijmegen, Paris VI) but the seminar showed new unexpected research directions (proof theory, constructive algebraic topology).

One outcome of this week was the decision to create a European group under the acronym MAP for "Mathematics: Algorithms and Proofs".

Abstracts of the talks (chronological order)

Henri Lombardi

Title: Dynamical algebraic structures, pointfree topological spaces and Hilbert's program

A possible relevant meaning of Hilbert's program is the following one: "give a semantic for classical mathematics". More precisely, give a systematic interpretation of classical abstract proofs (that use Third Excluded Middle and Choice) about abstract objects, as proofs about constructive versions of these objects.

If this program is fulfilled we are able "at the end of the tale" to extract constructive proofs of concrete results from classical abstract proofs of these results.

Dynamical algebraic structures or (this is more ore less the same thing) geometric theories seem to be a good tool for doing this job. In this setting, classical abstract objects are interpreted through incomplete concrete specifications of these objects.

The structure of axioms in geometric theories give rise in a natural way to distributive lattices.

The spectra of these lattices (as the Zariski spectrum or the real spectrum of a commutative ring) are, from a constructive point of view, pointfree topological spaces. Abstract objects correspond to classical points of these pointfree spaces.

We give some examples showing how all this "constructive rereading machinery" works when applied to classical abstract proofs in commutative algebra. E.g. when one uses local-global principles. Or when one uses the notion of Krull dimension: this notion is deciphered as a machinery of algebraic identites in the ring.

Herve Perdry

Title: Constructive Theory of Valued Fields

We first give a short general presentation about valued fields: Hensel's Lemma, Newton Polygon Algorithm, Henselian Fields.

Then we present brievely a general construction of the Henselization of a valued field, based upon successive formal adjunction of roots. The correctness of this construction is a consequence of the dynamical methods presented by Henri Lombardi in the previous talk.

Arjeh Cohen

Title: Group Theoretic Examples of Algorithms Providing Proof Certificates

Computer algebra has always had an emphasis on complexity of algorithms, so that bigger and bigger problems could be solved on a given machine. The internet will play an increasingly large role in the exchange of mathematics between people, and we believe this will require a different approach to computational mathematics. As the exchange of mathematics across the World Wide Web becomes easier than solving all problems locally, the management of mathematical queries becomes more prominent. The problem of verifying the correctness of computations is particularly acute when they are no longer done on local machines with software the user trusts.

By way of experiment, we have implemented eight group theoretic *queries*: invocations of permutation group algorithms that have been developed over the years and that are implemented as part of the computer algebra package GAP. The *response* to a query is the output of the algorithm, which may have been run on a remote computer which the user knows nothing about. The user has reason to doubt the validity of the response, and so will demand some kind of *verification*. Since our queries are of a mathematical nature, this verification should take the form of (an encoding of) a proof.

A classical example is the factorization of a natural number. If a sequence $p_1, p_2, \ldots, p_t$ of numbers is returned as a response to the query "factor the natural number n," it is easy for the user to verify whether $n = p_1 \cdots p_t$. In order to verify that each p_i is a prime number, it would be very useful to receive additional data, such as the primality witnesses for each p_i. This example has been worked out by Olga Caprotti, Martijn Oostdijk and the first author.

We treat computational permutation group theory in a similar manner. Our eight queries trigger responses which are either human readable proofs or the mathematical data (the *certificates*) needed to put together such a proof. The proofs could be transformed to a computer checkable proof without too much effort (in fact, this has been done in collaboration with Pollet and Sorge). The work on the eight queries, ranging from group membership to the order of a permutation group, is joint work with Scott Murray.

For more details, see http://www.win.tue.nl/ amc/pub/permgp.pdf

Francis Sergeraert

Title: Constructive Algebraic Topology

Some typical examples are used which show that the natural *constructive* aim is not covered in classical Algebraic Topologic.

Considering Algebraic Topology from this point of view led the lecturer and his colleague Julio Rubio to new versions of various exact and spectal sequences. These new versions are in particular *effective*, giving new *algorithms* allowing interested topologists to compute some homology and homotopy groups so far unreachable.

An important computing work has been undertaken along the lines so opened. The Kenzo program, a 16000 lines Lisp program, is now available implementing the theoretical ideas around the essential notion of *locally effective* object.

A small demonstration is proposed to concretely illustrate the difference between effective and locally effective objects, and showing the physical nature on a computer of an *object with effective homology*.

Julio Rubio

Title: Formal Analysis of Symbolic Computation systems for Algebraic Topology

The interest of using formal methods in the analysis, development and modelling of symbolic computation systems is briefly stated. This approach is then particularised to Sergeraert's systems as EAT (Effective Algebraic Topology) or Kenzo (see the talk by F. Sergeraert at this same Seminar). We focus on a particular case: the Basic Perturbation Lemma (or BPL, in short), which is one of the central components in the design of algorithms in Algebraic Topology and Homological Algebra.

The formal analysis, in this example and in general, is divided in two lines: algebraic specifications (joint work with L. Lambán, V. Pascual and C. Domínguez, from Universidad de La Rioja) and mechanised theorem proving (joint with C. Ballarin, from Technische Universität München, and J. Aransay, from Universidad de La Rioja).

By means of algebraic specifications, it can be proved that the EAT (or Kenzo) data structures are "as general as possible", since they are ingredients of final objects in suitable categories of Abstract Data Types implementations.

In the second line, the Isabelle proof assistant is used to give a mechanised proof of the BPL. To this aim, algebraic structures are encoded in Isabelle trough dependent sets and extensible records (see the talk by C. Ballarin at this same Seminar).

The slides of the talk are avaible at
http://www.unirioja.es/dptos/dmc/psycotrip/RubioAtDagstuhl.ppt

Clemens Ballarin

Title: Algebraic Structures in Isabelle/HOL

Reuse of algebraic (and in fact, any) theories in a proof assistant requires the proof language (script language) to provide some sort of module system. We present the approach taken in the Isabelle/HOL system, namely the use of *locales* and the explicit representation of algebraic structures as record types or dependent sets.

The creation of an algebraic base library for Isabelle/HOL serves two purposes:

- Evaluation of the module system
- Providing the necessaary theories for the mechanisation of the Basic Perturbation Lemma (see presentation by Julio Rubio).

Helmut Schwichtenberg

Title: Extracting Programs from Proofs

It is well known that it is undecidable in general whether a given program meets its specification. In contrast, it can be checked easily by a machine whether a formal proof is correct, and from a constructive proof one can automatically extract a corresponding program, which by its very construction is correct as well. This – at least in principle – opens a way to produce correct software, e.g. for safety-critical applications. Moreover, programs obtained from proofs are 'commented' in a rather extreme sense. Therefore it is easy to maintain them, and also to adapt them to particular situations.

The talk concentrates on the question of classical versus constructive proofs. It is known that any classical proof of a specification of the form $\forall x \exists y A(x, y)$ with $A(x, y)$ quantifier-free can be transformed into a constructive proof of the same formula. However, when it comes to extraction of a program form a proof obtained in this way, one easily ends up with a mess. Therefore, some refinements of the standard transformation are necessary.

In the lecture such refinements are developed, and some examples are studied in detail.

Gilles Dowek

Title: Preliminary investigations of induction over real numbers

The goal of this talk is to present a pinciple on real numbers similar to induction on natural numbers. We show that on several examples, proofs using this principle are simpler and more direct than proofs using an alternative principle such as the existence of a least upper bounds. We discuss the relation between this principle and ordinal induction and also how proofs using this principle can be reduced.

Ulrich Kohlenbach

Title: Proof Mining: A Logical Approach to Computational Mathematics

The first part of the talk gives a survey on how techniques from Mathematical Logic, so-called proof interpretations, can be used to extract new information from ineffective proofs in various areas of mathematics and, in particular, in functional analysis. In the second part we present the results (in part jointly with Laurentiu Leustean) of a recent case study where this approach has been applied to proofs in metric fixed point theory. This concerns the asymptotic regularity of various iteration schemes of nonexpansive functions. Our results, which extend to the general setting of convex metric spaces (Takahashi) resp. hyperbolic spaces (Goebel,Kirk,Reich) and to directionally nonexpansive functions (Kirk), not only provide new effective bounds but even yield systematically new qualitative results on the uniformity of asymptotic regularity. The latter generalize all known results of this kind which had been obtained by functional analytic embedding techniques during the last 20 years. We conclude the talk by presenting a new general logical meta-theorem which implies such uniformity results "a priori" if certain easy to check logical conditions are met. Only to get explicit effective bounds one has to carry out an actual proof interpretation.

The slides are at http://www.brics.dk/ kohlenb/dagstuhl03.pdf

Laureano Gonzalez-Vega

Title Computing algebraic certificates in real algebraic geometry

We show how difficult is to compute algebraic certificates for geometrical statements in Real Algebraic Geometry by using as main examples Finiteness Theorem and Pierce-Birkhoff Conjecture (is every piecewise polynomial continuous function from R^n to R a finite combination of sup, inf and polynomials?). In the first case it is shown explicitely how to compute the open description of the set of degree four univariate polynomials without real root quoting that the used technique is difficult to extend to more complicated situations. In the second case, it is shown how the algorithm provided by the rational solution for n=1 is still not known to be polynomial. (joint work with Henri Lombardi)

The web link to the slides is:
http://frisco.matesco.unican.es/ gvega/ficheros/daghstul.pdf

Dmitrii Pasechnik

Title: Complexity of semi-algebraic proofs

It is a known approach to translate propositional formulas into systems of polynomial inequalities and to consider proof systems for the latter ones. The well-studied proof systems of this kind are the Cutting Plane proof system (CP) utilizing linear inequalities and the Lovasz-Schrijver calculi (LS) utilizing quadratic inequalities. We introduce generalizations LS^d of LS that operate with polynomial inequalities of degree at most d.

It turns out that the obtained proof systems are very strong. We construct polynomial-size bounded degree LS^d proofs of the clique-coloring tautologies (which have no polynomial-size CP proofs), the symmetric knapsack problem (which has no bounded degree Positivstellensatz Calculus proofs), and Tseitin's tautologies (which are hard for many known proof systems). Extending our systems with a division rule yields a polynomial simulation of CP with polynomially bounded coefficients, while other extra rules further reduce the proof degrees for the aforementioned examples.

Finally, we prove lower bounds on Lovasz-Schrijver ranks and on the size and the "Boolean degree" of Positivstellensatz Calculus refutations. We use the latter bound to obtain an exponential lower bound on the size of static LS^d and tree-like LS^d refutations.

Loic Pottier

Title: Proofs of polynomial inequalities in Coq

In order to help proofs in real analysis, we have begun to implement a tactic which solves polynomial inequalities with real coefficients, and produce the complete proof of the solution, using the theory of types of the Coq system. This tactic has two parts: - first we adapt a method from Bochnak-Coste-Roy-Hörmander to compute, by euclidian divisions, all the possible signs of the involved polynomials. From these signs we can conclude for the existence of solution for the inequaliies. - second, we build from a trace of execution of the algorithm, a proof of the seult. This proof uses only polynomial equalities and applications of various forms of the intermediate value theorem. For the moment, the tactic is completely implementesd in one variable. The Hörmander method is implemented in the general case, and works in simple non trivial cases.

Serge Mechveliani

Title: Term rewriting for automated proofs in algebra and programming

We discuss the project of bringing automatic proof possibility to computer algebra systems. To our mind, term rewriting (TRW) technique should be very useful here. Also it is desirabe some adequate programming tool: AAS — any Appropriate Algebraic Specification language and tool for TRW (order sorted TRW logic, abstract theories, reflection, and so on).

Induction by appropriately chosen expressions combines naturally with TRW, making it fit to prove 'usual' theorems in algebra and programming.

Some particular features of the projects are pointed out, as partial completion and resource approach.

Certain simple first-approach strategy is introduced and a couple examples are solved with it, like (N+M = M+N) for natural numbers and reverse(reverse(Xs)) = Xs for lists. The talk

describes the recent state of study and investigation.

The slides are available at http://www.botik.ru/ mechvel/papers.html

Marc Daumas

Title: Formal Approach to Floating Point Numbers

I present in this talk the work initiated with Laurent Thery and Laurence Rideau and continuing with the PhD of Sylvie Boldo. Floating point arithmetic is heavily used in critical applications both off-line for the engineering and simulation of future designs and in-situ to control processes. Validating such applications typically incurs lots of testing and mathematical developments in numerical analysis. Most results in numerical analysis state that floating point is an approximation to real arithmetic where tiny relative round-off errors (perturbations) are introduced with each atomic operation. This approach has been very successful but a few catastrophic bugs have led people to refine this definition when it is possible and needed.

We have designed in Coq a formal specification that includes the IEEE standard floating point arithmetic. The talk presents some of our achievements and our feeling about formal methods. It starts with a brief survey of existing tools for formal verification and former specifications of floating point arithmetic. Achievements include the exact representation of the round-off error, the two's complement floating point arithmetic implanted in some DSP, the expansions of floating point numbers to produce multiple precision arithmetic and the faithful evaluation of polynomials with Horner's rule.

The slides are available at
http://www.ens-lyon.fr/ daumas/SoftArith/Dagsthul.pdf

Peter Paule

Title: Symbolic Summation: Constructive Aspects and Verification

The talk presents various thoughts on constructive aspects of computation and verification related to recent work in symbolic summation and special functions. Illustrative examples concern Zeilberger's paradigm (e.g., certificate proofs of definite sums evaluations via the derivation of linear recurrences with polynomial coefficients), d'Alembertian solutions to linear difference equations in difference fields (e.g., Schneider's extension of Karr's indefinite summation machinery), and closure properties of d-finite (holonomic) functions and sequences (e.g., the NIST-DLMF project of creating a new Digital Library of Mathematica Functions). Papers connected to the topics of the talk are to find at

http://www.risc.uni-linz.ac.at/research/combinat/risc/

David Delaye

Title: Dealing with Algebraic Expressions over a (Commutative) Field in Coq using Maple

We describe an interface between the Coq proof assistant and the Maple symbolic computation system, which mainly consists in importing, in Coq, Maple computations regarding algebraic expressions over (commutative) fields. This can be either pure computations, which do not require any validation, or computations used during proofs, which must be proved (to be

correct) within Coq. These correctness proofs are completed automatically thanks to the tactic `Field`, which deals with equalities over (commutative) fields. This tactic may generate side conditions (regarding the denominators), which must be proved by the user, and has been implemented in a reflexive way, which ensures both efficiency and certification. The implementation of this interface is quite light and can be very easily extended to get other Maple functions (additionally to the four functions we have imported and used in the examples we give). (joint work with Michaela Mayero)

Bernard Mourrain

Title: Symbolic Numeric Methods for Certified Computations

In this talk, we consider the problem of certification in geometry, from an effectivity point of view. We describe several examples, where approximate but certified computation are required when dealing with geometric objects such as curves and surfaces. Several methods combining algebraic, symbolic and numeric computation are described and illustrated on typical problems such as computing the arrengement of curves, surfaces, their topology, ... More details can be found in

http://www-sop.inria.fr/galaad/mourrain/Cours/20030106dagsthul.pdf

Wieb Bosma

Title: Certificates in Number Theory

As part of a project in Nijmegen to combine the strengths of 'proof assistants' and computer algebra systems we investigate the possibilities of using the results of calculations inside theorem provers. In this talk I considered one of the simplest cases, where the proof assistant would use the factorization of an integer in a sceptical way. This requires that the computer algebra system provides certificates for primality of the prime factors it exhibits. An overview was given of the state of the art of factorization and primality proving algorithms, as well as some results on certificates for compositeness and for primality. To illustrate the way these problems can be handled in the Magma computer algebra system, I have also shown an implementation of the Agarwal, Kayal, and Saxena algorithm. This exciting new algorithm, published in September 2002 was the first deterministic method for distinguishing primes from composite numbers that runs in polynomial time. One of the curious properties of this algorithm is that it is entirely elementary, and only the complexity bound required some hard analytic number theory. (However, analysis by Hendrik Lenstra has shown that even that can be largely overcome.)

The slides of the talk can be found at
http://www-math.sci.kun.nl/ bosma/PandA/talk.ps

Peter Schuster

Title: Ring Spectra Without Points

The purpose of this study is to pave the way from formal topology to algebraic geometry. In addition to casting prime and maximal ideals for a secondary part, we thus aim at a constructive road to algebraic geometry of as predicative a nature as possible. In return, the category of commutative rings is embedded into that of formal geometries.

To start with, the present formal version of the Zariski topology on the prime spectrum of a commutative ring is enriched with a coinductively generated positivity relation. A formal counterpart of the structure sheaf is then introduced that equally represents the given ring; this may further serve as a guiding example for a notion of sheaves on formal topologies not only of that particular kind.

We also invent formal geometries, a natural formalisation of the category of locally ringed spaces that allows to rephrase the universal property characteristic of the Zariski spectrum together with the aforementioned structure sheaf. In contrast to even the locale-theoretic approach, neither points nor stalks need to occur, let alone any invocation of Zorn's lemma.

Josef Schicho

Title: Ill-posed Problems in Computer Algebra

Traditionally, computer algebra computes with exact domains, which are not subject to approximation errors or roundoff errors. However, there are several situations where it is of advantage to study classical problems in computer algebra over approximative domains such as the floating point numbers. This leads typically to ill-posed numerical problems, where the answer does not depend continuously on the input parameters. Examples are the computation of GCDs, polynomial factorization, polynomial decomposition, rational parametrization. Instead of solving the ill-posed problem as it is, one is usually satisfied with the solution of a well-posed problem which is "nearby" - this is the approach of regularization. Interpreting the output by mathematically precise and verifiable statements is often highly nontrivial. The main purpose of this talk is to throw some light on these difficulties, hopefully leading to clarifying discussions.

Renaud Rioboo

Title: A presentation of the FoC project

This talk describes the current state of the FoC project. The project is a joint effort of researchers from the Laboratoire d'Informatique de Paris 6 (LIP6) of Université Pierre et Marie Curie, the Institut National de Recherche en Informatique et Automatique (INRIA) and the Conservatoire National des Arts et Métiers (CNAM). The purpose of the project, started in late 1997 is to provide tools for certified computer algebra. Following the Axiom initiative, we propose to offer to computer algebra developers a language suitable for both expressing and certifying computer algebra algorithms. A compiler translates user level code into Objective Caml code that runs efficiently and to Coq code that the user can certify in an interactive Coq session.

The slides are available at http://calfor.lip6.fr/ rr/dagstuhl.ps

Erik Palmgren

Title: Constructing order completions in type theory

The constructive real numbers are known to verify only a weakened form of the axioms for total order. It is a so-called pseudo-order. We examine two kinds of completions by cuts. For arbitrary dense pseudo-orders these are Dedekind cuts, and for divisible, pseudo-ordered

groups, we consider Cauchy cuts. We show how these can be predicatively defined, using a generalisation of dependent choice.

Jepser Carlstrom

Title: Descriptive Definitions in Type Theory

Descriptive definitions are very common in mathematics: you prove there is a unique x satisfying $P(x)$ and then give that x a name. For instance, it is common to define a^{-1} as 'the x such that $ax = 1 \wedge xa = 1$'.

In formalizing mathematics in type theory, one has to translate the descriptive definitions to explicit ones, because there is no support for descriptive definitions in type theory. There are wellknown translations but they are not useful in practice because they yield long and unnatural proofs.

I will give a very direct interpretation in type theory by translating a modified version of Sören Stenlund's natural deduction-style system for first order intuitionistic logic with descriptors. The interpretation has several advantages, among them being the fact that it seems to be useful in practice.

Bas Spitters

Title: Constructive Peter-Weyl's Theorem

We claim that, contrary to Weyl's belief, constructive mathematics suffices for the applications of mathematics. To support our claim we prove the Peter-Weyl theorem in a constructive and natural way. For this proof we need constructive integration theory, Gelfand theory and spectral theory. These theories will be outlined in the talk. As proposed by Weyl we stress that mathematics should be build on basic observables or finite approximations.

Participants

- Areski, Nait Abdallah (INRIA Le Chesnay)
- Ballarin, Clemens (TU München)
- Bosma, Wieb (Radboud University Nijmegen)
- Capretta, Venanzio (University of Ottawa)
- Carlström, Jesper (University of Stockholm)
- Cohen, Arjeh M. (TU Eindhoven)
- Coquand, Thierry (Chalmers – Göteborg)
- Daumas, Marc (ENS – Lyon)
- Delahaye, David (CNAM)
- Doligez, Damien (Université Paris VI)
- Dowek, Gilles (Ecole Polytechnique – Palaiseau)
- Gonzalez-Vega, Laureano (University of Cantabria)
- Kohlenbach, Ulrich (TU Darmstadt)
- Lombardi, Henri (Université de Franche-Comté)
- Mahboubi, Assia (INRIA Sophia Antipolis)
- Mechveliani, Sergey D. (Russian Academy of Science)
- Mourrain, Bernard (INRIA Sophia Antipolis)
- Palmgren, Erik (Uppsala University)
- Pasechnik, Dimitrii V. (Universität Frankfurt)
- Paule, Peter (Universität Linz)
- Perdry, Hervé (University of Pisa)
- Pottier, Loic (INRIA Sophia Antipolis)
- Rioboo, Renaud (UPMC – Paris)
- Roy, Marie-Françoise (Université de Rennes 1)
- Rubio-Garcia, Julio (Universidad de la Rioja)
- Schicho, Josef (Universität Linz)
- Schuster, Peter (LMU München)
- Schwichtenberg, Helmut (Universität München)
- Sergeraert, Francis (Université de Grenoble)
- Southall, Alan (Siemens AG – München)
- Spitters, Bas (Radboud University Nijmegen)