Algebraic, topological, and homotopical aspects of computer algebra and formalisation

Advanced Mathematics second year program
2022-2023, Lyon

Coordinated by
Philippe Malbos, Institut Camille Jordan (ICJ).

Lecturers
Yoann Dabrowski (ICJ), Florence Fauquant–Millet (ICJ), Stéphane Gaussent (ICJ), Kenji Iohara (ICJ), Olga Kravchenko (ICJ), Philippe Malbos (ICJ), Samuel Mimram (École Polytechnique), Filippo A. E. Nuccio Mortarino Majno di Capriglio (ICJ), Alistair Savage (Department of Mathematics & Statistics, University of Ottawa).

Presentation of the Program
Constructive mathematics, decidability theory, proof theory in computer science and mathematics, topological data analysis and formalisation of mathematics have led to recent developments of original categorical, homotopical and homological models of computations and proofs. This program aims to present a flavour of all these new approaches according to three axes: computational algebraic topology, diagrammatic categories and formalisation of mathematics. The objective is also to prepare students for research in these three very active areas of mathematics at the interplay with fundamental computer science.

The program consists in six courses: three basic courses at the fall semester and three advanced courses at the spring semester. The basic and advanced courses last 24 hours. Basic courses offer a broad introduction to these three area:

- Computational Algebraic Topology;
- Categorical representation theory;
- Introduction to homotopy type theory $(*)$.

In the advanced courses, the spectrum is deepened in all three areas, while maintaining links between the addressed topics. The themes covered in these courses will be directly related to current research:

- Computational Topology and Persistent Homology $(*)$;
- Operadic and diagrammatic linear rewriting $(*)$;
- Formalisation for the working mathematician.

Courses above marked with a star (∗) are designed to be accessible to both students of this course and to the ones of the Master 2 in fundamental computer science from École Normale Supérieur de Lyon. The advanced courses will particularly welcome the participation of PhD students and junior researchers.

The multiplicity of approaches and concepts around computational algebraic topology, theory of representations and formalisation of mathematics developed in this course will carried on in the possibilities of coaching Master 2 thesis in a wide variety of fields. The possibilities for further training after the program are both academic and non-academic, with a buoyant job market in companies for profiles in applied fundamental mathematics. Two courses had already been built in the same spirit and proposed on similar themes: *Representation Theory, from Schubert Calculus to Rewriting* 2019-2020 and *Topological and algebraic calculus* 2016-2017. Each time, it was a success in terms of students' job opportunities.

Basic courses

Computational Algebraic Topology (24 h) - Kenji Iohara and Olga Kravchenko.

Abstract. Mathematics studies many phenomena often with the same tools, looking for similar patterns in different areas of knowledge. This course is viewed as a panorama of such tools coming from algebraic topology, with certain examples of applications along the way. In particular, one of the objectives of this course is to give some geometric intuition to seemingly abstract general theory. Further applications of homotopy and (co)homology will be met in computations of invariants of manifolds. Connections between homology and cohomology theories, as well as homotopy and homology theories will be explored.

Tentative plan

- Axiomatic approach to (co)homology complexes. Categories and functors. Examples of (co)homology theories.
- Cup product. Künneth formula
- Duality theorems
- Classical applications of homotopy and (co)homology theories
- Spectral sequences. Examples
- Morse theory. Homotopy type from critical values. Gradient flow. Homotopy spheres. Morse inequalities

References

- H. Edelsbrunner, J. L. Harer, *Computational topology. An introduction.* American Mathematical Society, Providence, RI, 2010.
- A. Hatcher, *Algebraic Topology*, Cambridge University Press (2002). *Available at*
https://pi.math.cornell.edu/ hatcher/AT/ATpage.html

- J.McCleary, John *A user's guide to spectral sequences.* Second edition. Cambridge Studies in Advanced Mathematics, 58. Cambridge University Press, Cambridge, 2001.
- J. Milnor, *Morse theory.* Annals of Mathematics Studies, No. 51 Princeton University Press 1963.

Categorical representation theory (24 h) - Stéphane Gaussent and Alistair Savage.

Abstract. The theory of monoidal categories gives a powerful and intuitive set of tools for studying representation theory and other branches of algebra. It also leads to connections between representation theory and other fields of mathematics, such as low dimensional topology, knot theory, algebraic combinatorics, and mathematical physics.

We will begin by giving an introduction to monoidal categories, along with the string diagram calculus typically used in this field. Intuitively, this can be viewed as doing "two-dimensional algebra". The language of string diagrams helps us build geometric intuition and is also the framework upon which the applications to knot theory and topology are built.

We will next explain how various symmetry and duality properties of representations are encoded in simple string diagram relations. This allows one to construct categories describing representations of the classical Lie groups starting only from a knowledge of the symmetry that such categories posses.

Finally, we turn our attention to *categorification*, which is a way of revealing hidden structure in well-known algebras and their representations. We will discuss the general theory, and illustrate this theory with the examples of Heisenberg categories and Kac–Moody 2-categories. These categories help us to better understand the representation theory of the symmetric groups, Iwahor–Hecke algebras, Heisenberg algebras, Lie algebras, and quantized enveloping algebras.

Tentative plan

- Strict monoidal categories and string diagrams
- Duality and symmetry in monoidal categories
- Brauer categories and oriented Brauer categories
- Categorification
- Heisenberg categories
- Kac–Moody 2-categories

References

- A. Beliakova & A. D. Lauda, *Categorification and Higher Representation Theory*, Contemporary Mathematics, Volume 683 (2017), 361 pages.
- J. Brundan, *On the definition of Kac-Moody 2-category*, Math. Ann. 364 (2016), 353–372.
- A. Savage, *String diagrams and categorification*, preprint 2018, https://arxiv.org/abs/1806.06873.

Introduction to homotopy type theory (24 h) - Philippe Malbos and Samuel Mimram.

Abstract. This course introduces homotopy type theory which allows formalizing, in a synthetic way, properties of spaces which are invariant up to homotopy: types can be interpreted as spaces, and typed programs as constructions on these spaces.

This approach is based on the Curry-Howard correspondence between typed programs and proofs in natural deduction, which extends to the rich setting of dependent types as introduced by Martin-Löf, as well as the univalence axiom due to Voevodsky.

We will explain how many classical constructions in algebraic topology can be performed in this framework and we will put it to practice using the Agda proof assistant.

Tentative plan (each session is 2h CM + 2h TP)

- Intuitionistic type theory (cut-elimination, coherence), simply-typed lambda-calculus, the Curry-Howard correspondence
- Introduction to the Agda proof assistant
- Dependent types, identity types and basic properties (transport, groupoid laws, ...)
- n-types : propositions and logic (Diaconescu's theorem), sets (Hedberg's theorem), propositional truncation (the axiom of choice)
- Univalence : equivalences (various formulations), function extensionality
- Higher inductive types (circle, suspensions, truncation, quotients) in cubical Agda, the fundamental group of the circle, rewriting coherence theorems with applications in homotopy type theory.

References

- *Homotopy Type Theory*, The Univalent Foundations Program, IAS (2013). *Available at* https://homotopytypetheory.org/book/.
- S. Mimram, *Program = proof*, 2020, `https://www.lix.polytechnique.fr/Labo/Samuel.Mimram/teaching/INF551/course.pdf`.

Advanced courses

Computational Topology and Persistent Homology (24 h) - Kenji Iohara and Philippe Malbos.

Abstract. With the development of numerical tools and the prodigious increase in storage resources and computing power, the production of data is exploding in many fields of science, humanities and social sciences, engineering and health. These data are of very different kinds, but often voluminous and produced as vectors of very large dimensions. It is therefore very difficult to visualise them. Generally, very few coordinates are relevant to the questions we ask about these data. Also, it is necessary to eliminate useless information, even though we do not always know which coordinates are relevant to observe. It also happens that the data produced are noisy with missing information. The challenge is then to analyse qualitatively and quantitatively these data by taking into account all these problems of data production processes.

In this course we show how *persistent homology* (PH), a method from algebraic topology and Morse theory, can be applied to analyse data sets represented by point clouds equipped with a distance function. These point clouds must be considered as finite and noisy samples, taken from a geometric object.

The objective is then to study the topological properties of the geometric object by eliminating the small topological features corresponding to the noise. PH is a descriptive method from algorithmic topology that aims to qualitatively study the features on the geometric object that persist over several scales of observation. PH provides algorithms for 1/ to eliminate unnecessary information in the data, such as noise, 2/ to encode the evolution of the topology according to a sub-level decomposition of a filtration, and 3/ to measure the relative importance of the topological features according to the filtration. PH algorithms are robust to perturbations of the input data, regardless of their dimensions and coordinates. They compute compact representations of qualitative features of the input data called *barcodes*, which are very simple discrete structures and easy to compare. PH computing is a very open and growing field that presents many important and fascinating challenges. There are many developments, both in the research of new algorithms, their optimisations for computing on large datasets, and their software implementations, but also in its vast fields of applications. In particular, there are many applications in the field of biology, notably in geometric pattern recognition problems, with results in oncology, genetics, and the study of molecular features of protein structure.

Tentative plan

- Surfaces: two-dimensional manifolds, Searching a triangulation and simplification, morse functions;
- Simplicial complexes, filtrations, Čech complexes, Vietoris–Rips complexes;
- Persistent homology, stability theorems, efficient implementations, sparse matrix implementation;
- Generalised forms of persistence: multidimensional persistence, quivers and zigzags;
- Biological applications: persistent homology analysis of gene expressions and protein structures.

References

- G. Carlsson, *Topology and data*. Bull. Amer. Math. Soc. (N.S.), 46(2):255–308, 2009.
- H. Edelsbrunner, D. Letscher, and A. Zomorodian, *Topological persistence and simplification*. 28, 511–533, 2002. Discrete and computational geometry and graph drawing (Columbia, SC, 2001).
- H. Edelsbrunner and J. L. Harer, *Computational topology*. American Mathematical Society, Providence, RI, 2010.
- V. Kovacev–Nikolic, P. Bubenik, D. Nikolić, G. Heo, *Using persistent homology and dynamical distances to analyze protein binding*, Statistical Applications in Genetics and Molecular Biology, vol. 15, no. 1, 2016, pp. 19–38.
- J. R. Munkres, *Elements of algebraic topology*. Addison Wesley Publishing Company, Menlo Park, CA, 1984.
- K. Xia and G.-W.Wei, *Persistent homology analysis of protein structure, flexibility, and folding*. Int J Numer Method Biomed Eng, 30(8) :814–44.

Operadic and diagrammatic linear rewriting (24 h) - Stéphane Gaussent and Yoann Dabrowski.

Abstract. Rewriting in algebraic structures has appeared independently in several situations: to decide the word problem in monoids, to compute linear bases through normal forms or obtain algebraic freeness results from Gröbner bases of ideals in algebras and operads. In this lecture, two applications of the formalism of linear rewriting will be studied in higher linear structures such as operads and 2-categories.

Symmetric algebraic Operads are an algebraic tool encoding types of algebraic structures. For instance, associative, commutative associative or Lie algebras and operads themselves are encoded in such way as algebras over operads. Koszul operads are an effective tool to study (co)homological properties of their algebras. In the first part of this course, we will study the generalization of Gröbner bases to operads, through the algorithmic framework of linear rewriting. The symmetric group action presents an inherent difficulty, solved using shuffle operads introduced by Dotsenko and Khoroshkin. We will present their critical branching criteria of confluence and their Buchberger algorithm to compute Gröbner bases. We will show that these Gröbner bases allow to obtain a poincaré-birkhoff-witt criteria of Koszulity for symmetric quadratic operads.

The rewriting practiced in the context of polygraphs also makes it possible to study these presentations from a constructive and algorithmic point of view, making it possible to calculate invariants of these 2-categories, and bases of the spaces of morphisms. In this course, some techniques and computations of bases of space of morphisms will be performed.

Tentative plan

- Symmetric and Shuffle operads and their presentations.
- Operadic Grobner Basis in terms of operadic rewriting. Buchberger algorithm. Critical branching lemma.
- Application to Kozulness of operads.
- Polygraphic approach to diagrammatic linear categories
- Bases theorem in higher linear rewriting
- Computations of bases in the Kac-Moody 2-category

References

- M. Bremner & V. Dotsenko, *Algebraic Operads: An Algorithmic Companion* Chapman & Hall / CRC Press, 2016.
- V. Dotsenko & A. Khoroshkin, *Gröbner basis for Operads*, Duke Math. J. 153, 2 (2010) pp 363–396.
- B. Dupont *Rewriting modulo isotopies in Khovanov-Lauda-Rouquier's categorification of quantum groups*, Advances in Mathematics, Volume 378, Feb 2021.
- Y. Guiraud & P. Malbos, *Polygraphs of finite derivation type*, Mathematical Structures in Computer Science, Vol. 28, Issue 2, pp. 155–201, 2018.

Formalisation for the working mathematician (24 h) - Filippo Nuccio.

Abstract. The main goal of this course is to introduce `Lean-3` and to formalise the main properties of the fundamental group of a topological space in it. `Lean` is one of the so-called *theorem provers*, a software designed to let users code mathematics into a computer in order to formally verify the correctness of proofs. Several theorem provers exist, some of the most popular ones being `Agda`, `Coq`, `Isabelle`, `HOL`, `Mizar` and `Lean`. All these programs allow users to code definitions, to state theorems and to check their proofs. Although the process of entering mathematics into a computer, called *formalisation*, is different in each of these provers, they all share a similar architecture and most of them, including `Lean`, rely on dependent type theory and on the Calculus of Inductive Constructions. The course will allow students to move their first steps in the world of formalisation and to get acquainted with the challenges and the possibilities offered by this new branch of mathematics, while working on the concrete example of fundamental groups and applying the knowledge acquired in the introductory course on type theory.

The choice of using `Lean` is somewhat arbitrary, basically coming from the teacher's experience with the software and the very recent development of `mathlib`, its reach and deeply interconnected mathematical library. One of the major developments recently achieved in `Lean` is the formalisation of the main technical core of the preprint *Lectures on Analytic Geometry* by Clausen–Scholze, or the ongoing project of formalising Smale's result on the existence of a *sphere eversion*.

Tentative plan

- CM 1 (2h): Introduction to `Lean`: basics commands, tactics, fundamental examples (induction proofs, proofs as terms of a type, difference of set-theoretic membership and type-theoretic one);
- TP 1 (3h): Exercise session: definition of group, subgroup, normal subgroup, group homomorphism;
- TP2 (3h) : Exercise session: basic properties of topological spaces, continuous functions, homotopies;
- TP 3 (3h) : Exercise session: definition of $\pi_1(X, x)$ and first properties (probably without associativity);
- TP 4 (3h) : Exercise session: more advanced properties of $\pi_1(X, x)$;
- CM 2 (2h): Discussion of solutions to previous exercises and comparison with pre-existing definitions of groups, topological spaces and continuous functions in `mathlib`;
- TP 5 (3h) : Exercise session: definitions and first properties of coverings;
- TP 6 (3h) : Exercise sessions: formalisation of the Galois correspondence for coverings;
- CM 3 (2h): Conclusion and discussion of further directions.

References

- J. Avigad, L. de Moura and S. Kong, *Theorem Proving in Lean - Release 3.23.0*, October 2021. *Available at* https://leanprover.github.io/theorem_proving_in_lean/
- *Homotopy Type Theory*, The Univalent Foundations Program, IAS (2013). *Available at* https://homotopytypetheory.org/book/

- The mathlib Community *The lean mathematical library.* In *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs* (*CPP 2020*). Association for Computing Machinery, New York, NY, USA, 367–381 (2020). DOI: https://doi.org/10.1145/3372885.3373824