

Autumn school "Proof and Computation", 13th and 14th September 2021

An international autumn school "Proof and Computation" will be held virtually from 13th to 14th September 2021. Its aim is to bring together young researchers in the field of Foundations of Mathematics, Computer Science and Philosophy. Previous autumn schools in this series were [PC19 in Herrsching](#), [PC18 in Fischbachau \(Aurachhof\)](#), [PC17](#), [PC16](#).

Scope

- Predicative Foundations
- Constructive Mathematics and Type Theory
- Computation in Higher Types
- Extraction of Programs from Proofs

Talks

- Liron Cohen: Effectful Proving (and Disproving)
- David Corfield: Modal Type Theories
- Grigory Devadze: Computer-certified proofs for control theoretic aspects and program extraction
- Dominik Kirst: Formalising Metamathematics in Constructive Type Theory
- Klaus Mainzer: Proof and Computation: Perspectives of Mathematics, Computer Science, and Philosophy
- Antonio Piccolomini d'Aragona: Epistemic grounding. Semantics and calculi based on Prawitz's theory of grounds
- Sebastian Posur: On free abelian categories for theorem proving
- Florian Steinberg: Continuity, computability and discreteness
- Holger Thies: Complexity Theory in Analysis with Applications to Differential Equations

Schedule

- 13. September, 14-15 Mainzer
- 13. September, 15-16 Corfield
- 13. September, 16-17 Cohen
- 14. September, 09-10 Thies
- 14. September, 10-11 Steinberg
- 14. September, 11-12 Piccolomini
- 14. September, 14-15 Devadze
- 14. September, 15-16 Posur
- 14. September, 16-17 Kirst

Registration

If you are interested in participating please send an email to Nils Köpp ([koepp\[at\]math.lmu.de](mailto:koepp[at]math.lmu.de)).

Abstracts

- David Corfield: Modal Type Theories. Modal logic has a long history in philosophy stretching back to C. I. Lewis's work early in the last century. Over recent decades many researchers have looked to develop modal variants of type theory, with applications extending from computer science to mathematical physics. In this talk we will consider the ways in which philosophers might make use of modal types.
- Dominik Kirst: Formalising Metamathematics in Constructive Type Theory. In this talk, I outline our investigation of metamathematical results using constructive type theory as implemented in the Coq proof assistant. The first part of the talk introduces the synthetic approach to computability based on the fact that all functions definable in a constructive meta-theory are computable, hence avoiding reference to formal models of computation such as Turing machines. The second part demonstrates how this approach can be applied to some decision problems in first-order logic: validity and provability (the original Entscheidungsproblem), finite satisfiability (Trakhtenbrot's theorem), and entailment in first-order axiom systems such as PA and ZF (yielding incompleteness of PA and ZF). Complementing these negative results, I'll also briefly discuss the constructive analysis of the completeness of first-order logic in our setting.

- Klaus Mainzer: Computer science, logic, and mathematics are connected in the constructions of intuitionistic type theory. In this sense, mathematical constructivism seems to open new avenues of research combining foundations of logic and mathematics with highly topical challenges of IT- and AI-technology. References: K. Mainzer, *The Digital and the Real World. Computational Foundations of Mathematics, Science, Technology, and Philosophy*, World Scientific Singapore 2018; K. Mainzer/P. Schuster/H. Schwichtenberg (Eds.), *Proof and Computation. Digitization in Mathematics, Computer Science, and Philosophy*, World Scientific Singapore 2018; K. Mainzer/P. Schuster/H. Schwichtenberg (Eds.), *Proof and Computation II. From Proof Theory and Univalent Mathematics to Program Extraction and Verification*, World Scientific Singapore 2021.
- Antonio Piccolomini d'Aragona: Epistemic grounding. Semantics and calculi based on Prawitz's theory of grounds. Building upon Prawitz's *theory of grounds*, I propose a formal framework for epistemic grounding. First, I outline a "universe" of typed epistemic grounds over atomic bases over first-order background type-languages. Then, I introduce formal languages of grounding and define over them denotation functions that map terms onto epistemic grounds. I explore some properties of this denotational semantics - e.g. correctness, canonical closure, primitive/non-primitive extensions of a language of grounding etc. Finally, I define formal systems over languages of grounding for proving relevant facts about epistemic grounding - e.g. the given term has the given type, the given terms are computationally equivalent, computation instructions for evaluating terms to canonical forms are convergent etc. I prove that each system is sound and enjoys a normalisation property, and I investigate some consequences of these results.
- Sebastian Posur: On free abelian categories for theorem proving. Computing explicitly within a free mathematical object can be interpreted as theorem proving. In this talk, we discuss the constructiveness of free abelian categories. A very concrete description of free abelian categories was given by Murray Adelman, and we demonstrate how his description can be employed to validate homological lemmata like the Snake lemma computationally.
- Holger Thies: Complexity Theory in Analysis with Applications to Differential Equations. Computable analysis is an extension of classical computability theory to computations with uncountable entities such as the real numbers. The rigorous study of computational complexity of real functions based on this model was initiated by Ko and Friedman. The theory can be extended to more general spaces using the framework of representations. In this talk we introduce the necessary notions from computable analysis and real number complexity theory and present some applications to operators in analysis, in particular problems related to solving ordinary and partial differential equations. If time allows we also explore some connections to constructive mathematics and formalization in proof assistants.

The workshop is supported by the [Udo Keller Stiftung](#) (Hamburg) and the CID (Computing with Infinite Data) programme of the European Commission.

[Klaus Mainzer](#)

[Peter Schuster](#)

[Helmut Schwichtenberg](#)

Last change: 14. July 2021