Short-Term Scientific Mission Grant
- APPLICATION FORM[1] -

Action number: CA20111

Applicant name: Loïc Pujet

Details of the STSM

Title: Strictification of the syntax of type theory

Start and end date: 02/09/2024 to 06/09/2024

Goals of the STSM

Purpose and summary of the STSM.

Nowadays, a significant portion of the research in type theory is backed by formal proofs. But these formal proofs are hampered by the difficulty of dealing with the intricacies of type-theoretic syntax: researchers have to choose between the low-level manipulation of untyped syntax, or the complex equational reasoning that is required with typed syntax. The purpose of the STSM is to develop new methods that allow typed syntax to be used without the equational boilerplate, resulting in more streamlined and elegant proofs.

This project contributes directly to the development of the metatheory of type theory which is a goal of WG6: we develop methods for computer formalisation of type theories in modular, reusable ways, which allow reasoning about their syntax abstractly. Our first attempt will be to apply the technique to a concrete type theory, but we expect that the same techniques are applicable to any type theory.

Working Plan

Description of the work to be carried out by the applicant.

1 This form is part of the application for a grant to visit a host organisation located in a different country than the country of affiliation. It is submitted to the COST Action MC via e-COST. The Grant Awarding Coordinator coordinates the evaluation on behalf of the Action MC and informs the Grant Holder of the result of the evaluation for issuing the Grant Letter.

There are several ways to describe the syntax of type theory, from the concrete to the abstract: abstract syntax trees together with typing and conversion relations (extrinsic syntax), well-scoped syntax trees, well-typed syntax trees (intrinsic syntax), well-typed syntax quotiented by conversion. Recently it was discovered [1,2] that canonicity, normalisation or type checking can be implemented directly on the most abstract version of syntax, as e.g. normalisation preserves both typing and conversion, and there is no need to refer to untyped terms or break the quotient during these constructions. However, computer formalisations of normalisation [3,4] for type theory still use concrete low-level syntax because of the difficulties with working with intrinsic syntax in proof assistants: typing rules depend on conversion rules, this results in transport operations appearing on intrinsic terms, which results in extreme amount of boilerplate when working with intrinsic terms. This problem is usually referred to as "transport hell".

In this project we propose a solution via strictification: by making some equations in the syntax definitional, there is no need to transport over them anymore, removing the transport-boilerplate. We plan to use Pédrot's strict presheaf model [5] to strictify the substitution calculus part of the syntax, and prove the an induction principle for this stricter syntax. We will to showcase that this technology makes it possible to formalise metatheoretic proofs about type theory without the extra boilerplate.

The host is an expert in intrinsic formalisations, the guest is an expert of Pédrot's model.

[1] Thorsten Altenkirch, Ambrus Kaposi: Normalisation by Evaluation for Dependent Types. FSCD 2016: 6:1-6:16

[2] Thierry Coquand: Canonicity and normalization for dependent type theory. Theor. Comput. Sci. 777: 184-191 (2019)

[3] Andreas Abel, Joakim Öhman, Andrea Vezzosi: Decidability of conversion for type theory in type theory. Proc. ACM Program. Lang. 2(POPL): 23:1-23:29 (2018)

[4] Arthur Adjedj, Meven Lennon-Bertrand, Kenji Maillard, Pierre-Marie Pédrot, Loïc Pujet: Martin-Löf à la Coq. CPP 2024: 230-245

[5] Pierre-Marie Pédrot: Russian Constructivism in a Prefascist Theory. LICS 2020: 782-794

Expected outputs and contribution to the Action MoU objectives and deliverables.

Main expected results and their contribution to the progress towards the Action objectives (either research coordination and/or capacity building objectives) and deliverables.

The STSM contributes directly to objective 7 of the Action MoU (Develop a modular theory of type theories). So far, a large part of the progress on that front has has been centred around the development of abstract forms of syntax (categories with families, etc) in order to relate type theories to each other and to study the metatheory of type theory in a modular way [6]. Our project provides an efficient method to *implement* syntax based on categories with families in a proof assistant, in order to bring modularity to the formalised metatheory of type theory.

We expect to be able to deliver an Agda formalisation of the initial model of Martin-Löf type theory that uses Pédrot's strict presheaves. The initiality of the model should provide us with an induction principle, that can in turn be used to give streamlined proofs of many important metatheoretical results, such as normalisation and canonicity. This is a direct contribution towards deliverable 15 of the Action MoU (Prototype implementation of the mathematical framework for modular reasoning about type theories and their extensions).

The host and the guest have already started collaborating on an Agda formalisation, and our initial attempts, which seem promising, are publicly available at https://bitbucket.org/akaposi/psh .

[6] : Rafaël Bocquet, Ambrus Kaposi, Christian Sattler: For the Metatheory of Type Theory, Internal Sconing Is Enough. FSCD 2023