Integrating Cost and Behavior in Type Theory

Robert Harper
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

The computational view of intuitionistic dependent type theory is as an intrinsic logic of (functional) programs in which types are viewed as specifications of their behavior. Equational reasoning is particularly relevant in the functional case, where correctness can be formulated as equality between two implementations of the same behavior. Besides behavior, it is also important to specify and verify the cost of programs, measured in terms of their resource usage, with respect to both sequential and parallel evaluation. Although program cost can – and has been – verified in type theory using an extrinsic formulation of programs as data objects, what we seek here is, instead, an intrinsic account within type theory itself.

In this talk we discuss Calf, the Cost-Aware Logical Framework, which is an extension of dependent call-by-push-value type theory that provides an intrinsic account of both parallel and sequential resource usage for a variety of problem-specific measures of cost. Thus, for example, it is possible to prove that insertion sort and merge sort are equal as regards behavior, but differ in terms of the number of comparisons required to achieve the same results. But how can equal functions have different cost? To provide an intrinsic account of both intensional and extensional properties of programs, we make use of Sterling's notion of Synthetic Tait Computability, a generalization of Tait's method originally developed for the study of higher type theory.

In STC the concept of a "phase" plays a central role: originally as the distinction between the syntactic and semantic aspects of a computability structure, but more recently applied to the formulation of type theories for program modules and for information flow properties of programs. In Calf we distinguish two phases, the intensional and extensional, which differ as regards the significance of cost accounting – extensionally it is neglected, intensionally it is of paramount importance. Thus, in the extensional phase insertion sort and merge sort are equal, but in the intensional phase they are distinct, and indeed one is proved to have optimal behavior as regards comparisons, and the other not. Importantly, both phases are needed in a cost verification – the proof of the complexity of an algorithm usually relies on aspects of its correctness.

We will provide an overview of Calf itself, and of its application in the verification of the cost and behavior of a variety of programs. So far we have been able to verify cost bounds on Euclid's Algorithm, amortized bounds on batched queues, parallel cost bounds on a joinable form of red-black trees, and the equivalence and cost of the aforementioned sorting methods. In a companion paper at this meeting Grodin and I develop an account of amortization that relates the standard inductive view of instruction seequences with the coinductive view of data structures characterized by the same operations. In ongoing work we are extending the base of verified deterministic algorithms to those taught in the undergraduate parallel algorithms course at Carnegie Mellon, and are extending Calf itself to account for probabilistic methods, which are also used in that course.

(This talk represents joint work with Yue Niu, Harrison Grodin, and Jon Sterling.)

2012 ACM Subject Classification Theory of computation → Type theory; Theory of computation → Design and analysis of algorithms; Theory of computation → Logic and verification

Keywords and phrases type theory, analysis of algorithms, program verification

Digital Object Identifier 10.4230/LIPIcs.CALCO.2023.1

Category Invited Talk

Related Version *ACM POPL 2022*: `https://doi.org/10.1145/3498670`

Supplementary Material *Software*: `https://github.com/jonsterling/agda-calf`
archived at `swh:1:dir:790153bb2b1e4c1fa942283307af0ebe070d14a8`

10th Conference on Algebra and Coalgebra in Computer Science (CALCO 2023).
Editors: Paolo Baldan and Valeria de Paiva; Article No. 1; pp. 1:1–1:2
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Funding This research was sponsored by the Air Force Office of Scientific Research (AFOSR) under award number A210038S0002 and the National Science Foundation under award number CCF1901381. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.