Interaction Tree Specifications: A Framework for Specifying Recursive, Effectful Computations That Supports Auto-Active Verification (Artifact)

Lucas Silver
University of Pennsylvania, Philadelphia, PA, USA

Eddy Westbrook
Galois, Inc., Portland, OR, USA

Matthew Yacavone
Galois, Inc., Portland, OR, USA

Ryan Scott
Galois, Inc., Portland, OR, USA

Abstract

This paper presents a specification framework for monadic, recursive, interactive programs that supports auto-active verification, an approach that combines user-provided guidance with automatic verification techniques. This verification tool is designed to have the flexibility of a manual approach to verification along with the usability benefits of automatic approaches. We accomplish this by augmenting Interaction Trees, a Coq datastructure for representing effectful computations, with logical quantifier events. We show that this yields a language of specifications that are easy to understand, automatable, and are powerful enough to handle properties that involve non-termination. Our framework is implemented as a library in Coq. We demonstrate the effectiveness of this framework by verifying real, low-level code.

2012 ACM Subject Classification Theory of computation → Denotational semantics; Theory of computation → Programming logic; Theory of computation → Separation logic

Keywords and phrases coinduction, specification, verification, monads

Digital Object Identifier 10.4230/DARTS.9.2.8

Related Article Lucas Silver, Eddy Westbrook, Matthew Yacavone, and Ryan Scott, "Interaction Tree Specifications: A Framework for Specifying Recursive, Effectful Computations That Supports Auto-Active Verification", in 37th European Conference on Object-Oriented Programming (ECOOP 2023), LIPIcs, Vol. 263, pp. 30:1–30:26, 2023. https://doi.org/10.4230/LIPIcs.ECOOP.2023.30

Related Conference 37th European Conference on Object-Oriented Programming (ECOOP 2023), July 17–21, 2023, Seattle, Washington, United States

Evaluation Policy The artifact has been evaluated as described in the ECOOP 2023 Call for Artifacts and the ACM Artifact Review and Badging Policy.

1 Scope

This artifact formalizes the definitions and theorems presented in the associated paper in the Coq proof assistant.

2 Content

Definitions and verified theorem proofs are contained in the provided codebase. The file Artifact_README.md provides exhaustive mappings from names and identifiers in the paper to names in the code.

Dagstuhl Artifacts Series
Schloss Dagstuhl – Leibniz-Zentrum für Informatik,
Dagstuhl Publishing, Germany

3 Getting the artifact

The artifact endorsed by the Artifact Evaluation Committee is available free of charge on the Dagstuhl Research Online Publication Server (DROPS). In addition, the artifact is also available at: `https://github.com/GaloisInc/entree-specs`. And a Docker image is provided at: `https://zenodo.org/record/7423277`.

4 Tested platforms

This codebase should build on any system with the following dependencies:

- coq >= 8.15
- coq-paco >= 4.1.2
- coq-itree >= 5.0.0

5 License

The artifact is available under license Creative Commons Atribution 4.0 International.

6 MD5 sum of the artifact

8b2b29dad8ea433e43604d15e8358e58

7 Size of the artifact

1.5 GiB