

Certified Compilation of Skeletal Semantics

Alan Schmitt

September 17, 2018

keywords Formal semantics, programming languages, compilation, certification

1 Context

Skeletal semantics¹ are a new approach to describe the semantics of programming languages in a highly generic and modular way. These semantics describe the control flow for the evaluation of each construct of the language without specifying the implementation of primitive construct. For instance, the semantics for the addition of two expressions could be "evaluate the first expression, then evaluate the second expression, then add the results", without specifying how to add numbers. One may then provide several *interpretations* of such semantics, such as a big-step semantics, a static semantics (i.e., typing), or an abstract semantics (for abstract interpretation). These interpretations are defined independently of the programming language, and thus may be reused for other languages.

2 Topic

The objective of the internship is to study the certified compilation of skeletal semantics. The student will choose a source language, for instance a simple WHILE language, and a target language, for instance a stack-based language. The student will define these languages as skeletal semantics, and will define the compilation from source to target as an interpretation of the source semantics. The student will then establish a set of necessary conditions between the behavior of primitive constructs to certify the compilation is

¹<http://people.rennes.inria.fr/Alan.Schmitt/papers/skeletal.pdf>

correct: the behavior of a compiled program corresponds to the behavior of the source program.

This internship will take place in the Celtique team (<https://team.inria.fr/celtique/>) at Inria Rennes, in France. It will be supervised by Alan Schmitt (alan.schmitt@inria.fr). The student is expected to have notions on the semantics of programming languages and of compilation.

3 Bibliography

- Martin Bodin, Thomas Jensen, and Alan Schmitt. 2015. Certified Abstract Interpretation with Pretty-Big-Step Semantics. In *Proceedings of the 2015 ACM Conference on Certified Programs and Proofs (CPP'15)*. ACM, 29–40.
- Xavier Leroy. 2009. Formal verification of a realistic compiler. In *Communications of the ACM*, 52(7):107–115.
- Gordon Plotkin. 1981. A Structural Approach to Operational Semantics. Technical Report FN-19. DAIMI, Aarhus University.
- David A. Schmidt. 1995. Natural-semantics-based Abstract Interpretation (preliminary version). In SAS. Springer LNCS vol. 983, 1–18.