Project proposal
Master Informatique

Inferring loop invariants for graph transformations

Keywords: Logic; Semantics of programming languages; Program verification; Abstract interpretation

Scientific Context

Within the Climt project[1], we are working on graph transformations, and in particular the verification of graph transformations.

Graph transformations have wide-spread applications, such as pointer-manipulating programs and model transformations in languages such as UML. They are also interesting for graph databases, such as those represented with RDF schemas, and the link structure of the World-Wide Web.

We are currently developing an imperative language for graph transformations, and a family of Hoare-style logics for reasoning about programs of this language. The logics are variants of Description Logics. A distinctive feature of our work is that both the programming language and the logic have been formalized in the proof assistant Isabelle[2].

Planned work

Our graph transformation language is an imperative language with typical constructors such as conditionals and loops. For reasoning about the correctness of loops, these have to be annotated with invariants by the programmer. The aim of this project is to automatically infer invariants for the existing language, as far as possible. These investigations might also lead to a proposal of other loop-like constructs (iterators, maps) or restrictions (*e.g.* on the operations of the loop body) that may lead to a restriction of expressivity but facilitate inferring invariants.

The work will consist of a *conceptual phase*:

- Study of the existing graph transformation language and its semantics.
- Study of existing techniques for inferring loop invariants (abstract interpretation, CEGAR).
- Proposal of method of inferring loop invariants.
- Possibly: proposal of modifications to the graph transformation language.

This phase will be followed by an *implementation phase*:

[1] www.irit.fr/~Martin.Strecker/CLIMT/

[2] http://isabelle.informatik.tu-muenchen.de/

- implementation of the inference method.
- possibly: integration into the program proof framework: does the inference of invariants provide insight into the way the correctness of the invariant can be proved, or possibly make a correctness proof superfluous?
- in the latter case, possibly: integration in the formal Isabelle development, by means of proof certificates.

Prerequisites and interests

From the participating students, we expect

- interest in formal semantics of programming languages, logics, methods of correctness proofs for imperative programs (*required*)
- experience with functional (Ocaml, Haskell) or OO-functional (Scala) languages (*required*)
- experience with interactive proof assistants (such as Coq, Isabelle, PVS) (*would be a plus*)

Administrative Context

Interested? Then please contact us for further inquiries and for applying for the project. The internship can be carried out at either of the sites of the Climt project:

- LIG (Grenoble), contact Rachid Echahed[3]
- IRIT (Toulouse), contact Martin Strecker[4]

Funding will be provided for the duration of the project. The project is an ideal starting point for a PhD thesis.

[3] http://membres-lig.imag.fr/echahed/

[4] http://www.irit.fr/~Martin.Strecker/