Proposition de Stage de Recherche M2R : Terminaison de programmes à structures de données complexes : combinaison de méthodes

Laure Gonnord – Maître de conférences Université Lyon1/LIP

Keywords Static Analysis, Term Rewriting, Shape Analysis, Separation Logic.

Context Proving the termination of a flowchart program can be done by exhibiting a ranking function, i.e., a function from the program states to a well-founded set, which strictly decreases at each program step. In [1], we proposed a general algorithm for the computation of linear ranking functions, which is an adaptation of the Linear-Programming based scheduling method of [2].

However, the limits of our approach are raised as soon as we are faced with recursive programs, or irregular data. In fact, the underlying model of program, namely, the *polyhedral model*, has intrinsic restrictions that have only been partially solved.

Internship subject In this internship, we will focus on the relationship between termination techniques and sequential scheduling/termination techniques. Indeed, the rewriting community has also studied the problem of termination :

- Automated termination analysis for *term rewrite systems (TRSs)* [3] is a well-studied topic and has given rise to several powerful, fully automatic tools over the last years (e.g. AProVE, MuTerm, TTT2). TRSs are particularly suitable to represent composite data structures, such as lists or trees.
- Recently, two-stage approaches to termination analysis, harnessing the power of termination tools for TRSs also for programming languages, have been proposed [4, 5] : In the first stage, symbolic execution and abstraction on programming language level are used to over-approximate all possible program executions. From this program analysis, one then extracts a TRS whose termination implies the termination of the original program. In the second stage, termination of this TRS is then analyzed with existing tools.

In this internship, we aim at finding another way to prove termination of sequential programs with data structures (without translating into TRSs).

The candidate is expected to :

- Read the state of the art research articles : sequential termination, rewriting termination.
- Propose a framework that is able to deal with sequential programs with inductively defined data-structures and recursive programs.
- Implement a prototype.

Références

[1] Christophe Alias, Alain Darte, Paul Feautrier, and Laure Gonnord. Multi-dimensional Rankings, Program Termination, and Complexity Bounds of Flowchart Programs. In *Static Analysis Symposium*, Perpignan France, 2010.

[2] Alain Darte and Frédéric Vivien. Optimal fine and medium grain parallelism detection in polyhedral reduced dependence graphs. *International Journal of Parallel Programming*, 25(6) :447–496, December 1997.

[3] Jürgen Giesl, Thomas Ströder, Peter Schneider-Kamp, Fabian Emmes, and Carsten Fuhs. Symbolic evaluation graphs and term rewriting : A general methodology for analyzing logic programs. In *Proc. PPDP '12*, pages 1–12, 2012.

[4] Jürgen Giesl, Matthias Raffelsieper, Peter Schneider-Kamp, Stephan Swiderski, and Réne Thiemann. Automated termination proofs for Haskell by term rewriting. *TOPLAS*, 33(2) :7 :1–7 :39, 2011.

[5] Carsten Otto, Marc Brockschmidt, Christian von Essen, and Jürgen Giesl. Automated termination analysis of java bytecode by term rewriting. In *RTA '10*, pages 259–276, 2010.

LIP – UMR CNRS / ENS Lyon / UCB Lyon 1 / INRIA – 69007 Lyon
Contact E-mail : Laure.Gonnord@ens-lyon.fr