Automatic verification of programs with complex data structures

Laboratory: LIAFA (CNRS, Université Paris Diderot)
Responsables: Constantin Enea and Mihaela Sighireanu
Contacts: {cenea,sighirea}@liafa.univ-paris-diderot.fr

Although the software verification has made important progress during the last ten years, the verification of programs manipulating dynamic memory and complex data structures is still a challenge for the research in this domain. Automatic reasoning about the behaviours of such programs is a challenging problem because the classical techniques (static analysis or model checking) face big scalability issues: the use of the dynamic memory leads to potentially infinite state models. Thus, the specification formalisms used (i.e., Hoare’s logics) has to be able to describe important properties of the dynamic memory and also to allow compositional reasoning which is a key for scalability.

The Separation Logic (SL) introduced by Reynolds and O’Hearn [Rey02, BCO04] has such expressiveness and local reasoning properties. However, the entailment checking problem for SL is undecidable in general. Recently, several fragments of SL have been identified to have a decidable entailment checking procedure and still be able to specify programs using interesting data structures, e.g., singly and doubly linked lists [CHO+11, PWZ13], nested and overlaid lists [ESS13], and some kind of trees [IRv13]. The techniques used by these decision procedures are various: graph homomorphism in [CHO+11, ESS13], reduction to first order logic with reachability and set constraints in [PWZ13], reduction to monadic second order logic in [IRv13]. On other side, efficient techniques based on tree automata have been proposed to specify and reason about programs with dynamic data structures [HHR+12].

This internship has as goal to identify a fragment of SL for which the techniques based on graph homomorphism and tree automata may be combined to obtain an efficient decision procedure for the entailment checking problem. This fragment shall include the ones proposed for linked lists (singly, doubly linked and nested) and some restricted kind of trees. The decision procedure may be implemented inside the Celia toolset developed at LIAFA, in the “Modeling and Verification” team.

References


