Département d'Informatique de l'Ecole Normale Supérieure

Separation Logic with Summary Predicates Described by Set Universal Quantification

Internship location : École Normale Supérieure ; 45, rue d'Ulm ; 75 230, PARIS.

Team : Équipe Sémantique et Interprétation Abstraite / Équipe-Projet "ANTIQUE".

Advisor & Contact : Xavier RIVAL (*e-mail* : `rival@di.ens.fr`, tél : 01 44 32 21 50, fax : 01 44 32 21 51)

Internship topic :

Shape analyses such as [1,2] aim at computing precise structural invariants over memory states. For instance, they can infer properties of linked data-structures, and help verifying safety properties such as absence of memory errors or the preservation of global structural invariants. They use separation logic [3] in order to describe memory states : they describe separate memory regions with logical predicates that either very precisely define the contents or memory cells, or summarize them using high level recursive predicates. Such predicates can describe structures such as linked lists and binary trees. Recent works [4] extended such analyses with set predicates to describe structures with sharing, provided they have at least one recursive backbone (the best example is a graph with a representation based on adjacency lists).

However, they cannot cope well with data-structures that do not have any form of recursive pattern. Examples include *graphs without adjacency lists* and *union find* structures.

The purpose of this internship is to study another form of summarization based on arbitrary sets of memory regions, that will be abstracted separately.

To achieve this, the internship will consider the following tasks :

1. **Definition and formalization of an extension to separation logics to express summary predicates defined by universal quantification over a set**. This task consists in defining the new logical connector and the general form of logical predicates that the analysis will manipulate together with the associated abstraction relation (the tie between program states and logical formulas that describe them). Moreover, this will also be the opportunity to check that the logics allows to write paper proofs for programs manipulating structures such as union finds or graphs without adjacency lists.
2. **Definition of the abstract domain operations**. The second task aims at defining the core analysis algorithms. Analyses based on inductive predicates [1,2] need to *unfold* inductive predicates to materialize cells and analyze updates precisely, and to *fold* inductive predicates in order to guarantee the termination of the analysis. The counterpart for these unfolding and folding mechanism, using the set of logical predicates fixed in the first point need to be defined, formalized, and proved correct.
3. **Implementation and evaluation**. The final step consists in the implementation and evaluation of the analysis. This implementation will be done in the context of the **MemCAD** static analyzer [5], so as to reuse the definition of basic logical predicates and analysis algorithms, and implement only the new constructions and algorithms defined in the two previous points.

Pré-requis :

For this internship, it would be preferable that the student is familiar enough with abstract interpretation techniques as taught in lecture "2–6 Abstract Interpretation : Application to Verification and Static Analysis".

Références

[1] Dino Distefano, Peter W. O'Hearn, Hongseok Yang. A Local Shape Analysis Based on Separation Logic. In TACAS'06, pages 287-302, 2006.

[2] Bor-Yuh Evan Chang and Xavier Rival. Relational inductive shape analysis. In POPL'08, pages 247–260, 2008.

[3] John C. Reynolds. Separation Logic : A Logic for Shared Mutable Data Structures. In LICS'02, pages 55-74, 2002.

[4] Huisong Li, Xavier Rival, Bor-Yuh Evan Chang. Shape Analysis for Unstructured Sharing. SAS'15, pages 90-108, 2015.

[5] MemCAD static analyzer. `https://www.di.ens.fr/~rival/memcad.html`