Département d'Informatique de l'Ecole Normale Supérieure

Separation Logic-based Abstraction, Three Valued Logic Abstraction and Reduction

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 [1,2,3] 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.

Two main families of shape analyses have been studied : on one hand, analyses based on three-valued logics like TVLA [1] use conjunctions of basic memory predicates in three valued logic to describe heaps ; on the other hand, analyses based on separation logic [4] like [2,3], let summarizing predicates (usually based on recursive definitions for structures with an inductive form) describe separate memory regions. These two families of analyses come with different families of strengths and weaknesses. Therefore an interesting approach consist in combining both abstractions in a single analysis tool, to exploit synergies between the two logics although their definitions are quite dissimilar. The most natural way for abstract domains to communicate information is reduction [5].

The purpose of this internship is thus to explore the combination of these two families of analyses into a unique static analysis. Several benefits are expected :

- the resulting analysis should be more expressive than each family of analyses, and should thus allow to verify more programs ;
- in particular, the TVLA basic predicates can help refining higher level inductive definitions in separating logics ;
- in the other hand, separation logic predicate can sometimes express in a more concise manner predicates that are complex to describe with TVLA basic predicates.

The expected tasks are the following :

1. **Formalize a core TVLA abstract domain**. In this phase, a basic set of TVLA predicates will be chosen and formalized, and turn into a basic memory abstract domain, that can be integrated into the **MemCAD** static analyzer [6].
2. **Reduction operation on basic logical predicates**. The second task consists in setting up a basic reduction function between basic TVLA predicates and basic region predicates. This will let abstract domains based on separation logics communicate effectively with the abstract domain based on three valued logics. This work should result into more precise static analyses.
3. **Compilation of inductive definition into basic memory predicates and back**. The last phase aims at a more ambitious reduction, that will help exchanging complex information between domains. We will consider a static analysis based on separation logics with inductive predicates. To achieve a reduction operation across such predicates, a form of compilation of inductive predicates into basic logical predicates will be required.

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] Shmuel Sagiv, Thomas W. Reps, Reinhard Wilhelm. Parametric Shape Analysis via 3-Valued Logic. In POPL'99, pages 105-118, 1999.

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

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

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

[5] Patrick Cousot, Radhia Cousot. Systematic Design of Program Analysis Frameworks. In POPL'79, pages 269-282, 1979.

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