Generalised Verification for Quantified Permissions

Master Thesis Project Description

Nadja Müller

Supervised by Alexander Summers, Prof. Dr. Peter Müller
Department of Computer Science
ETH Zürich

April 25, 2016

1 Description

This project is intended to extend Viper [1], a verification infrastructure for permission-based reasoning. It includes several front-ends which translate given programs to Viper's intermediate language Silver. Verification runs on Silicon, which is based on symbolic execution, and Carbon, a verification condition generator.

There are currently three features of Silver handling unbounded heap structures. Using predicates, it can handle data structures which are recursively defined, magic wands [2] can be used to keep track of partial data structure and quantified permissions [3] allows us to express pointwise specifications.

The support of quantified permissions is still limited. In Viper, the structure of a quantified permissions is defined as:

$$\textbf{forall}\ x : T :: c(x) ==> \textbf{acc}(e(x).f, p(x)), \tag{1}$$

where *c(x)* is a boolean expression,*e(x)* an injective reference-typed expression and *p(x)* a permission expression.

The goal of this project is to generalise this form and make the expression of quantified permissions as liberal as possible. New extensions we consider in this project are to further allow combinations of multiple quantified permissions, mixing quantified permissions with pure quantifiers, as well as supporting interactions with other Viper features such as predicates and magic wands.

2 Core Goals

The core goals for this project are to extend the support of quantified permissions. In particular, we want to design an approach to handle:

- nested quantified permissions
- combinations of quantified pure and permission-based assertions
- support of predicates within quantified permissions
- support of magic wands within quantified permissions

Additionally we aim to implement following features both in Silicon and Carbon, as well as test their functionality and performance:

- combinations of quantified pure and permission-based assertions
- support of predicates within quantified permissions

3 Extensions

As an extensions, the following features could additionally be implemented and tested:

- support of magic wands within quantified permissions in Carbon
- support of magic wands within quantified permissions in Silicon
- nested quantified permissions in Carbon
- nested quantified permissions in Silicon

References

[1] Peter Müller, Malte Schwerhoff and Alexander J. Summers. Viper: A Verification Infrastructure for Permission-Based Reasoning. 2016.

[2] Malte Schwerhoff and Alexander J. Summers. Lightweight Support for Magic Wands in an Automatic Verifier. 2015.

[3] Peter Müller, Malte Schwerhoff and Alexander J. Summers. Automatic Verification of Iterated Separating Conjunctions using Symbolic Execution. 2016.