Proof Obligations for Correctness of Modifies Clauses

Ralf Sasse

Overview

- What are modifies clauses used for?
- What are modifies clauses exactly?
- Correctness of modifies clauses.
- How to check correctness of modifies clauses automatically?
- Some more details on the use and checking of modifies clauses.

Use of Modifies Clauses

Modifies clauses are expected to be correct and are not checked:

- Program Verification Using Change Information by P.H. Schmitt and B. Beckert enhances the performance of KeY.
- Lemma Generation like in B. Katz's "Studienarbeit"
- ESC/Java, only uses modifies clauses.

Other approaches to check modifies clauses:

- Chase, a static checker, neither sound nor complete
- Spoto/Poll claim a sound and complete method for JML assignable clauses, but no implementation available.

Modifies Clauses

Modifies Clauses are sets of ground terms containing

- the variables which may change
- the attributes which may change where $o.att$ is the attribute att of the object that o refers to in the pre-state.

Modifies clauses are associated with either

- programs
- methods.

Everything which is mentioned in the modifies clause may be changed after the method call.

Example

DEMO

Correct Modifies Clause

With a given set of classes a modifies clause for a method is correct with respect to that given set of classes and the method it belongs to iff

- every variable and attribute which has changed after the method call is a member of the modifies clause.

It is not necessary for elements of the modifies clause to change. The correctness is non-modular and depends on the rest of the classes.

Example continued

Correct modifies clauses are:

modifies $= \{\ self.p;\ self.q.r\ \}$

modifies $= \{\ self.p;\ self.q.r;\ self.q\ \}$

An incorrect modifies clause is for example:

modifies $= \{\ self.q.r\ \}$

modifies $= \{\ \ \}$

Proof Obligation Generation

We want a method to generate proof obligations for KeY which

- can use the KeY prover to verify the modifies clause by verifying the generated proof obligation
- generates the proof obligations automatically.

Proof Obligation Formula

$$\bigwedge_{class \in set-of-appearing-classes}$$

$$\forall o : class$$

$$[\bigwedge_{att \in attributes-of-class}([\bigwedge_{mod \in fitting-modifies} o \neq mod]$$

$$\rightarrow \forall x(x = o.att \rightarrow < self.m() > x = o.att))]$$

Example continued

DEMO

Proof interaction necessary

Even though KeY's prover is used user interaction with the proof is necessary most of the time.
If the proof can't be closed, i.e. the modifies clause is not correct, the parts of the proof which can't be closed give good hints to what is missing in the modifies clause.

Implementation

- Automatic proof obligation generation works.
- User can decide whether the modifies clause gets checked.
- Modifies clause doesn't have to be given for every method, only for those where we want to have the advantages given by the methods using them.

Summary

- KeY can check modifies clauses for their correctness.
- No other implementation known which checks correctness.