Paper Review《Symbolically Computing Most-Precise Abstract Operations for Shape Analysis》

Paper Info

G.Yorsh, T.Reps, and M.Sagiv

TACAS 2004

Main Contribution

This work is inspired by TVLA. The authors employ a decision procedure for the logic used to express properties of data structures, which is the shape information. This is the popular way in the shape analysis. Based on TVLA, the SPASS theorem prover is used to solve the constraints and get the shape info after the execution of a single statement. The precondition filters the input shape structures expressed in the logical formula just like a pipe.

The main contributions include:

- Express the shape information in 3-valued logic.
- Borrow the idea of FOCUS operation in TVLA and propose the assume algorithm, which supports the best transformer construction.
- Some tricks are proposed in this work, including the use of undecidable logics and the pruning operation in the materialization.

Technical Details

The paper is based on the work of TVLA. Three-valued logic is used to express the shape information. In order to take advantages of this kind of logic, materialization is needed in the framework, which splits the state with the logical value 1/2 into two states(0 and 1). The remaining problem is to equip the domain of three-valued logical structures to the standard form of abstract interpretation. It is critically important to define the best transformer for three-valued logical structures.

In some conventional instances of shape analysis, only one shape graph is used to express the shape information at a specific program location. In this work, a set of shape graphs are used instead of a single one. These graph, named as the canonical abstraction graphs, are all the more concrete sub-instances. This approach can make the analysis more accurate.

The assume algorithm takes a set of canonical abstraction graphs and a specific condition as its inputs, and return the set of canonical graphs which satisfy the condition. The specific condition is the predicate on the three-valued logic structures, determined by the semantics of the statement. The returned graphs are the results of the materialization. The materialization starts with 1/2 for all of the nullary predicates and then repeatedly refines instances of 1/2 into 0 and 1. Because the three-valued logic structures are bounded, the

number of different structures that can be produced is finite, which guarantees that this process terminates.

In order to accelerate the algorithm, the shapes which don't satisfy the condition are left out before the materialization. Some undecidable logics such as monadic 2-nd order logic can be utilized in this work with some strategies of approximation including adding the time-out threshold for theorem prover.

Future Work

The best transformers are dependent on the semantics of the statements. They describe the effects of the statements on the shape structures. It is still a problem to extract the predicate on the two-store vocabulary determined by each statement. The automatic method is on demand if we want to achieve the shape analysis in practice. I have not found a explicit solution to this problem in this paper.