RHEINISCH-
WESTFÄLISCHE
TECHNISCHE
HOCHSCHULE
AACHEN

LEHRSTUHL FÜR INFORMATIK 2

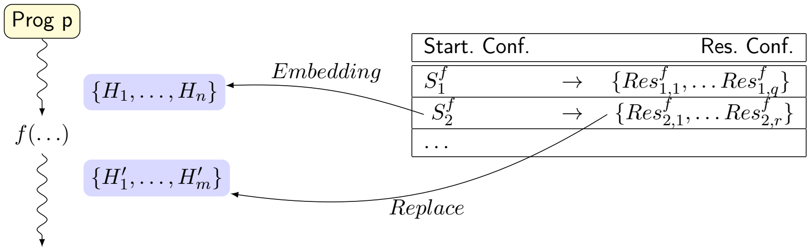
RWTH Aachen · D-52056 Aachen

Bachelor/Master Thesis

Interprocedural Heap-Analysis using Contracts

What is it all about?

Many software projects employ object-oriented languages, which introduce new challenges to formal reasoning. In particular, the analysis of programs that use pointers to implement dynamic data structures is a highly challenging and important task, as memory leaks or dereferencing null pointers can cause great damage especially when software reliability is at stake. As objects can be created at runtime, dynamic data structures possibly induce infinite heap state spaces and therefore cannot be handled by standard verification algorithms. Thus we employ graph-based abstraction to represent a potentially infinite set of states in a finite manner. This abstraction is then incorporated into a statical data-flow analysis generating a finite representation of the potentially infinite state space. Now in the presence of recursive procedures abstraction of heap states is not sufficient, as additionally the call stack may grow unboundedly. To tackle this problem the data-flow analysis can be extended to generate so-called procedure contracts. These contracts capture the results of procedure calls in an input-output-relation, i.e. whenever a procedure call appears and the current state of the heap matches the requirements of this contract (input), the heap state can be modified according to the contract's output without the need to (re-)analyse the called procedure.



Start. Conf.		Res. Conf.
S^f_1	$\rightarrow$	$\{Res^f_{1,1}, \dots Res^f_{1,q}\}$
S^f_2	$\rightarrow$	$\{Res^f_{2,1}, \dots Res^f_{2,r}\}$
$\dots$		

What has to be done?

The goal of this thesis is to incorporate the interprocedural heap-analysis for recursive programs into the present tool Juggrnaut. Moreover there is the possibility of developing theoretical background on this interprocedural heap-analysis using contracts such as proving soundness or refining contract information. In detail, this thesis involves the following steps:

- Obtaining a good understanding of the employed interprocedural analysis and corresponding heap abstraction.
- Implementation of the contract-generating analysis.
- Integration of the contract information into the analysis present in the Juggrnaut-tool.
- Optionally (Master): Proving soundness of the developed analysis, refinement of contracts

The thesis can be written in English or German.

Contact

- Christina Jansen, i2, E1 4205, (christina.jansen@cs.rwth-aachen.de)
- Thomas Noll, i2, E1 4211, (noll@cs.rwth-aachen.de)