DISJUNCTION ON DEMAND
MASTER'S THESIS PROPOSAL

DOMINIK GABI

1. Introduction

Abstract Interpretation describes a very general framework for static analyses. The framework is concerned with two domains: the *concrete* mathematical structure of interest (e.g. the semantics of a computer program) and some *abstract* approximation of that domain describing properties of interest. It states the necessary conditions for the two domains and the soundness of the abstraction/concretization relations between them, which allow us to draw conclusions about the structure based on the possibly much simpler abstraction.

The static analysis problem in general is *undecidable*. In practice, this means that we usually analyze some over-approximation of the semantics of a program. The most common way is to look at an over-approximation of the set of all reachable states. The price that has to be paid comes in terms of precision of the analysis. All information about how a state is reached is lost.

The paper "Trace Partitioning in Abstract Interpretation Based Static Analyzers" [1] addresses this problem by providing a flexible theoretical framework that makes the discrimination of traces possible. Instead of looking at reachable states, the analysis looks at partitions of traces leading to a state.

Furthermore, the authors present their implementation of the trace partitioning and demonstrate that it is applicable to real industrial software.

2. Project Proposal

The purpose of this thesis is to introduce this trace partitioning mechanism into *Sample* (Static Analysis of Multiple LanguagEs), a generic static analysis tool based on abstract interpretation that is currently being developed at ETH. Furthermore, interesting partitions should be investigated, e.g. based on the heap structure of a program.

The main tasks of the thesis will be the following.

- Formalize the partitions, i.e. find representation that can be passed to *Sample*.
- Figure out how to combine abstract traces from different partitions.
- Investigate interesting partitions.

3. Preliminary Schedule

The official deadline for the master's thesis is the first of September 2011. Until then, I plan to spend my time roughly along the following lines.

- Initial presentation of the problem statement.
- **2 months**: Analysis and design.
- 2nd presentation.
- **2 months**: Implementation.
- **2 months**: Documentation and write up.
- Final presentation.

Given the nature of the problem it seems reasonable to spend more time on the analysis and the design than on the actual implementation.

References

1. Laurent Mauborgne and Xavier Rival, *Trace partitioning in abstract interpretation based static analyzers*, European Symposium on Programming (ESOP'05) (M. Sagiv, ed.), Lecture Notes in Computer Science, vol. 3444, Springer-Verlag, 2005, pp. 5–20.