Type-base flow analysis: from polymorphic subtyping to CFL-reachability

Authors: Jakob Rehof , Manuel Fähndrich

Year: 2001

Paper Meta Data

Conference: POPL (Principles of Programming Languages)

Year: 2001

Number of pages: 13

Citations: 112

Paper Layout

1. Introduction
    - 1.1 Object Language
    - 1.2 Annotated types
2. Flow Analysis with polymorphic subtyping
    - 2.1 Problem 1: Constraint copying
    - 2.2 Problem 2: Demand-driven flow computation
    - 2.3 New method based on instantiation constraints
    - 2.4 CFL reachability
3. Polymorphism subtyping with instantiation constraints

4. Flow Relation
    - 4.1 Flow logic
    - 4.2 CFL formulation
    - 4.3 Examples
    - 4.4 Soundness

5. Algorithm
    - ...

6. Polymorphic and recursive type structure

7. Related work

8. Conclusion

What is the study about

Problem:

- The program analysis problems: data flow, type inference
- Interprocedural context sensitive analysis is used
- Current algorithms works slow [*O(n^8)*] and requires a lot of memory

Contribution of this paper:

- Using another well-known problem: Context-Free Language Reachability
- Improves algorithm [*O(n^3)*]
- Novel approach for polymorphic subtyping (*instantiation constraint*)
- This approach allows you to solve on-demand problems

Advantages and Disadvantages

Advantages:

- Clear contribution
- Formal wording and proofs
- Some necessary examples

Disadvantages:

- High entry threshold (Regular Grammar, Context Free Languages, ML language)
- Little explanation and a lot of links to other articles