

Title	別名解析に基づく静的単一代入形式変換アルゴリズムの実装と比較
Author(s)	西本, 真介
Citation	
Issue Date	2004-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/1798">http://hdl.handle.net/10119/1798</a>
Rights	
Description	Supervisor:片山 卓也, 情報科学研究科, 修士

# 別名解析に基づく静的単一代入形式 変換アルゴリズムの実装と比較

西本 真介 (210067)

北陸先端科学技術大学院大学 情報科学研究科

2003 年 2 月 13 日

キーワード: 静的単一代入形式 (SSA 形式), 別名解析, コンパイラ・フレームワーク, XML.

## 背景

静的単一代入形式 (Static Single Assignment Form) は, プログラム中の変数の使用に対して定義が一箇所だけになるように表された中間表現である. 静的単一代入形式を用いることで最適化の実現容易性と実行効率が向上するといわれている.

SSA 形式による最適化を行うためにはそれぞれのコンパイラで使用されている中間表現を SSA 形式に変換する必要がある. 代表的な SSA 形式の変換アルゴリズムとして Cytron らの方法 [1] と Sreedhar [2] らの方法がある. SSA 形式への変換では, ポインタの存在はデータフロー解析を難しくするために問題となる. 実際に, 従来の静的単一代入形式変換のアルゴリズムではポインタを使用することはできなかった.

この問題を解決するべく, ポインタ解析の結果を利用して静的単一代入形式変換を行うアルゴリズムが提案されている. 多くのアルゴリズムがフロー依存の解析を用いて変換を行う. Cytron らによるポインタ解析の方法 [3] もその中の一つである. それに対し, 最近では Hasti らがフロー非依存の解析を元にしたアルゴリズム [4] も提案している. 一般にフロー依存の解析はフロー非依存の解析に比べて得られる情報の精度が高いが, その分時間と空間を多く必要とするという特徴がある.

これらのアルゴリズムはその提案者によって実装が行われているものはあるが, 実際に同じコンパイラ・フレームワーク上でその性能が比較されてはいない. 例えば中谷の論文 [5] によると, 計算量の異なる二つの SSA 変換アルゴリズムが, 特殊な場合を除けばほとんど性能に差がないことを述べている. このように理論上の性能は必ずしも実用上の性能を意味しないことがある.

## 目的

本研究の目的は SSA 形式の変換時にポインタ解析を行うアルゴリズムを効率良く比較するための評価環境として, XML を利用したコンパイラ・フレームワークを作成することである. 実装した評価環境上で Cytron らの方法 [3] と Hasti らの方法 [4] の実装を行うことで評価環境の有用性を確認し, 最終的にはこれらのアルゴリズムの実用上の比較を行う.

## 結論

本研究ではポインタ解析に基づく SSA 変換のアルゴリズムの比較を目的として, XML による中間表現を用いたコンパイラ・フレームワークを実装した. 次にポインタ解析に基づく SSA 変換のアルゴリズムとしてフロー依存の解析を用いる Cytron らの方法 [3] とフロー非依存の解析を用いる Hasti らの方法の実装 [4] を行い, コンパイラ・フレームワークが次の利点を持つことを確認した.

- ポインタ解析に基づく SSA 変換のアルゴリズムを扱うのに十分な機能を持っている.
- XML による中間表現は実装コストを下げる.
- 実際にコンパイル, 実行を行って性能を測定できる.

しかし, これらのアルゴリズムの比較はまだ十分ではなく, 本研究で行った実験はアルゴリズムの比較を行うための予備的な位置付けであると考えられる. 今後, 更に改良を重ねて詳細な実験を行えるようにする必要がある.

## 参考文献

- [1] Ron Cytron, Jeanne Ferrante, Barry K. Rosen, Mark N. Wegman, and F. Kenneth Zadeck. An Efficient Method of Computing Static Single Assignment Form. Proceedings of the 16th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, pages 25-25, January 1989.
- [2] Vugranam C. Sreedhar and Guang R. Gao. A Linear Time Algorithm for Placing  $\phi$ -Nodes. Proceedings of the 22nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, pages 62-73, January 1995.
- [3] Ron Cytron and Reid Gershbein. Efficient accommodation of may-alias information in SSA form. Proceedings of the Conference on Programming Language Design and Implementation, pages 36-45, June 1993.

- [4] Rebecca Hasti and Susan Horwitz. Using static single assignment form to improve flow-insensitive pointer analysis. Proceedings of the ACM SIGPLAN '98 Conference on Programming Language Design and Implementation, pages 97-105, 1998.
- [5] 中谷俊晴. コンパイラ・インフラストラクチャにおける静的単一代入形式変換器の実装と評価. 東京工業大学 情報科学科 学士論文研究, 2001.