
発表概要

Flow Analytic Type System for Array Bound Checks

松野 裕[†] 佐藤 周行^{††}

型理論のコンパイラ設計およびプログラム解析における重要性は広く認識されつつあるが、実際のコンパイラなどに適用されるには多くの問題が残されている。根本的な問題として、アセンブラ言語などの低級言語にどのような型システムが対応するか、明らかではないことがある。この問題を解決するためにはデータフロー解析と融合された型理論が必要である。1 つには従来低級言語を解析するためにデータフロー解析を用いられてきたからであり、また低級言語のための型理論に関する既存研究では、明確に主張されていなくとも、本質的にはデータフロー解析を用いていると思われる研究も多く、さらにはデータフロー解析を適用することにより型理論が簡略化可能になる研究も少なくないからである。本発表ではデータフロー解析より導き出される型理論の枠組みを提案する。この枠組みにより、型理論による精密で、厳密な証明に依拠した解析、およびデータフロー解析による効率的な解析が可能となる。この枠組みを用いて配列の範囲チェックの解析のための型理論を提案する。特にコンパイラの最適化においてよく用いられながら、従来の型理論による研究では重視されてこなかった SSA 形式や Induction Variableなどを型理論に組み込むことを目指す。

Flow Analytic Type System for Array Bound Checks

YUTAKA MATSUNO[†] and HIROYUKI SATO^{††}

It is widely recognized that type systems are key theories for compiler design and program analysis. However, there remains much to be done for applying them to practical use. One problem is that it is not trivial what kind of type systems is appropriate for low level languages. To solve the problem, we believe that the type systems must be closely related to data flow analysis because it has been the major method for analyzing low level languages. Many previous works of type systems for low level languages essentially use data flow analysis though they did not address explicitly, or their systems would be much simpler if they use data flow analysis more directly. In this paper, taking analyzing array bound checks as an example, first we propose a framework for type systems for low level languages derived from data flow analysis. Second, we propose type systems for analyzing programs as a network of blocks (especially loops), dealing with SSA form and induction variables.

(平成 14 年 10 月 19 日発表)

[†] 東京大学大学院新領域創成科学研究科基盤情報学専攻
Department of Frontier Informatics, The University of
Tokyo

^{††} 東京大学情報基盤センター
Information Technology Center, The University of
Tokyo