

発表概要

変数条件に基づく原始帰納関数の再帰融合について

湯 浅 能 史†

本発表では原始再帰的な関数に関するプログラム変換の一手法について述べる。P を原始再帰関数とし Q を再帰子としたとき、合成関数 $Q*P$ において P の生成する中間データ構造を除去して Q を P に取り込むことは「再帰計算の融合」として知られている。例えば自然数の加乗算における結合法則や指数法則はこの典型例としてとらえることができる。再帰計算の融合は展開・畳み込み法によって実現可能であるが、本講演ではこれと同値でかつより洗練された形式を持つプログラム変換を提示する。素朴な展開・畳み込み法では、プログラム P (融合される側) 中の再帰部分をまず展開し、その結果得られたプログラムに畳み込み可能な構造が現れている場合に畳み込みを実行する。しかし、そうでない場合には既に行なった展開を遡ってキャンセルするという手順上の冗長性がある。また変換手続きの記述も複雑で理論的な分析の対象としては不都合である。一方本講演で与える手法では、変換を適用するに先立ってプログラム中の変数の現れ方を調べ、確実に畳み込みが成功する再帰部分を特定する。続く変換ではその箇所のみ展開を実行すればよい。畳み込み可能性の判定には「強自由変数」とよぶ新概念を利用するが、その定義はプログラムの構成に関する単純な帰納法で記述することができ、理論的な解析にも有用である。

A Fusion Procedure for Primitive Recursive Functions
Based on a Variable Condition

TAKAFUMI YUASA†

We present a way of transforming a program which calculates a primitive recursive function. It enables us to fuse a recursor Q into a program P automatically, by eliminating an intermediate data structure generated by P then consumed by Q. Our transformation is based on the unfolding-folding procedure, but it is not the aim of this work to describe the procedure itself in a naive way. With the unfolding-folding procedure, one may think it redundant that he has to try unfolding a recursion even if he will never fold it again. So, we will show instead a better way of the transformation, by which one can predict by analyzing variables in a program whether he will success or fail in folding a recursion, and then can unfold them only in the successful cases. Another advantage of our method is the refined formulation of it. The transformation is defined by a simple induction on the construction of a program, then is adequate to a theoretical analysis, while a naive formulation of the unfolding-folding procedure is so complicated that we can hardly analyze it.

(平成 11 年 3 月 24 日発表)

† 早稲田大学理工学総合研究センター
Advanced Research Institute for Science and Engineering,
Waseda University