

早稲田大学大学院理工学研究科

博士論文概要

論文題目

Studies on Recursion Removal Methods
in Program Transformation
プログラム変換における再帰除去法の研究

		申請者	
氏名		Yusuke	Ichikawa
		市川	祐輔

専攻・研究指導
(課程内のみ)

情報・ネットワーク専攻 並列知識情報処理研究

2007年 3月

高度情報処理社会の発展に伴い、多種多様な環境において個々の適性に合わせたソフトウェアが大量かつ迅速に必要とされている。そのため、ソフトウェアの生産性を向上させ、適切なソフトウェアを迅速かつ大量に供給するためのソフトウェア開発が急務である。このソフトウェアの開発の生産性は多くの要因に依存するが、中でもプログラムの形式は、プログラマーが表現する手段であるため、ソフトウェアの生産性に大きな影響力を与える。そしてプログラム言語の中で、再帰形式で表現される再帰プログラムは、多様な非定型的処理を直感的に表現できるため、一般的にソフトウェアの生産性が高くなるといわれている。そのため、ゲーム用ソフトウェアないし産業用ソフトウェアなどの一品生産的なソフトウェア開発においては、再帰形式が活用されている。

ところが、再帰プログラムによるソフトウェア開発は、開発段階における生産性が高い半面、再帰プログラムの実行がノイマン型計算機の基本原則との間にギャップがあるため、再帰プログラムはプログラムの効率的な実行や最適化の観点から問題のあるプログラム形式とされてきた。そこで、この問題に対する解決策として、再帰除去法というプログラム変換手法が提案されている。これは再帰プログラムを、自動的に高速な反復プログラムに変換する手法であり、再帰プログラムの高い生産性という利点を維持しつつ、高速な実行という最適化の効果を得ることができる。

本論文では、プログラム変換の一手法である再帰除去法として、累積関数法を提案する。再帰除去法とは、再帰プログラムを、計算量及び使用スペースを増加させず、スタック使用せずに反復プログラムに変換することである。再帰除去により、再帰呼出のオーバーヘッドが除去できるため、変換後の反復プログラムは変換前の再帰プログラムに比べ、高速に実行可能なプログラムとなる。

しかしながら、再帰プログラムは常に再帰除去が可能ではなく、再帰除去が困難であることも多い。そのため、現実のコンパイラにおいても実装されている再帰除去法は数少ない。そこで、再帰除去の研究においては、再帰除去が可能な範囲の再帰プログラムを拡大しつつ、できる限り高速化を行うという方針で研究が行われている。

そのような研究の中で、本論文では、累積関数法という再帰除去法を提案した。累積関数法は、指数関数の計算量を持つある種の非線形再帰プログラムを、線形の計算量を持つ反復プログラム、対数関数の計算量を持つ反復プログラム、さらにほぼ定数の反復のない閉式に変換する手法である。累積関数法は、指数関数の計算量をもつプログラムを、計算量を減少させたプログラムに変換しているため、その効果は大きく、入力によっては、数百以上の倍の速さで実行可能なプログラムに変換されている。

累積関数法の大きな特徴として、4点あげられる。

まず一つ目は、前述したように再帰除去法である点である。

二つ目は、重複計算を除去することができる点である。変換過程で計算量を減少させることができるのは、この重複計算の除去を行っているからである。一般にソフトウェアの開発においては、ボトルネックとなる部分を除去することが全体の性能向上に役立つとされているが、累積関数法は、この思想に沿ってボトルネックとなる重複計算の除去に主眼を置いた手法である。

三つ目は、変換後の反復プログラムがトップダウンに計算するという点である。トップダウンというのは、再帰除去前の再帰プログラムと同方向（再帰呼び出しの引数に対する計算の方向が同方向）という意味である。提案されている従来の再帰除去のほとんどが、再帰除去後の反復プログラムがボトムアップ（再帰呼び出しの引数に対する計算の方向が逆方向）であるため、本手法はそれらとは異なっている。ボトムアップに計算する利点は、再帰呼び出し内での引数への関数（後継関数）に、逆関数が存在することを仮定する必要がない、という点にある。すなわち、従来の再帰除去法によるボトムアップの反復プログラムは、後継関数に逆関数がない場合には、後継関数の計算量が増加するあるいはその値を保持すべきスペース量が増加するという問題点が生じる。他方、累積関数法では、後継関数に逆関数が存在しなくても、計算量およびスペース量を増加させない反復プログラムに返還できる。

四つ目は、実装が可能な程度に自動化可能な手法という点である。複雑な変換過程を経る再帰除去法の中には、実現が困難な手法もあるが、累積関数法は、実際に実装を行い、自動化の可能な再帰除去法であることを示した。これは、現実に利用されることの多い数論的プログラムの利便性を向上するために、数式処理システムの Mathematica 上において実装を行った。

次に本論文の全体構成を述べる。

第一章では、まず研究の背景について述べる。本論文で提案する累積関数法は、再帰除去法であり、かつプログラム変換法の一つである。そこで、ソフトウェアの生産性を向上するという広い視点から、プログラム変換の必要性及びその特徴、そして再帰除去法の必要性と特徴などについて述べる。

第二章から、本稿で提案する累積関数法を説明する。累積関数法とは、一定の条件を満たす非線形再帰プログラムに対して、累積関数がある場合に、変換が行われる。その変換は、累積関数定理によって線形反復プログラムに変換され、さらに累積関数定理の系 1 により対数関数反復プログラム、系 2 により反復のない閉式に変換する手法である。第二章では、この一定の条件を満たす非線形再帰プログラムについて、3 つのクラスを説明し、その具体的な 5 つの形式について定義する。そしてそれらに対応した累積関数を定義し、さらに累積関数定理を示してその証明を行う。また、累積関数定理の系について示し証明する。

第三章では、累積関数法の具体的な適用例を説明する。まずは、プログラム変換の分野で一般的に例題として使用されている、フィボナッチ関数を例題とする。

指数関数の計算量を持つフィボナッチ関数は、条件を満たし、かつ累積関数を持つため、累積関数法により線形の反復プログラムに変換され、さらに系 1・2 により、対数関数の計算量を持つ反復プログラム及び閉式に変換するステップを示す。次に累積関数法の特徴であるトップダウンの再帰除去法の効果を示すために、Double-Hailstorm 関数を例にとって説明する。これは、後継関数に逆関数が存在しないため、従来の再帰除去法では、後継関数の計算に二乗の計算量が必要となる。しかし、累積関数法では計算量を増加させず、必要なスペース量も増加させずに再帰除去が可能であることを示す。さらに、ドミノ問題を対象とした変換を示す。これは計算機科学の代表的な書籍である Concrete Mathematics に掲載されている問題の一つであり、母関数の解法の例題として紹介されている。累積関数法では、ドミノ問題を線形反復プログラム・対数関数反復プログラムに変換することに加え、上記記載の閉式と数学的に同値の閉式に自動的に変換できることを示す。これは高度な数式処理システムにより、ドミノ問題と同種の漸化式を母関数を用いずに自動的に変換できることを示す。

第四章では、リスト型再帰プログラムを対象とした累積関数法について述べる。まず flatten 関数などのリスト型再帰プログラムに対して、擬似結合性を用いた再帰除去法が提案されているが、累積関数法においても、同様に再帰除去ができることを示す。そして非線形再帰プログラムに対する累積関数法の適用を説明し、ハノイの塔のパズルを例題として累積関数法を適用する。ここでは、破壊的関数を明示的に用いることで、指数関数のメモリ量と指数関数の計算量が必要であるハノイの塔のパズルを、線形のメモリ量で線形の計算量で実行される反復プログラムに変換可能であることを示す。

第五章では、累積関数法の実装を述べる。累積関数法はコンパイラなどソフトウェア開発の場で使用されることが主眼であるが、再帰プログラムの有用性から一般のソフトウェア上においても必要とされるケースが多い。数式処理システムでは、数論的再帰プログラムを直感的に再帰プログラムで記述できるため、それらを実行すべく、数式処理システムの Mathematica 上で実装した。本章では、その実装の詳細について述べ、その効果を示す。

第六章では、関連研究について述べる。プログラム変換は過去 30 年間に渡り種々の観点から研究されているため、累積関数法と関係の深い幾つかのプログラム変換手法及び再帰除去法を説明し、その上で累積関数法の位置づけを明確にする。さらに、累積関数法と適用対象が類似しているタプリング法との関係について、実験を交えて詳細に比較した。その検証により、タプリング法と累積関数法は相補的な関係にあり、コンパイラの実装においては累積関数法をまず適用しその後タプリングを適用するのが最善策との結論を得た。

最後に第七章において、本論文を総括し、累積関数法の今後の研究課題について概観する。

研 究 業 績

種 類 別	題名、 発表・発行掲載誌名、 発表・発行年月、 連名者（申請者含む）
論文	<p>後継関数を持つリスト型非線形再帰プログラムに対する再帰除去法. 情報科学技術レターズ, Vol.5, pp.9-12, 2006年9月. 市川祐輔, 上田和紀.</p> <p>Recursion Removal from Recursive Programs with Only One Descent Function. IEICE Trans. Inf. & Syst., Vol.E88-D. No.2, pp.187-196, February 2005. Yusuke Ichikawa, Zenjiro Konishi, Yoshihiko Futamura.</p> <p>Unified approach to Recursion Removal, Logarithmic Iteration, and Loop Elimination for a class of Non-Linear Recursive Programs. Proc. International Workshop on Program Understanding, pp.18-28, July 2003. Yusuke Ichikawa, Zenjiro Konishi, Yoshihiko Futamura.</p> <p>単一後継関数を持つ再帰プログラムからの再帰除去及び閉式化. 情報科学技術レターズ, Vol.1, pp.27-28. 2002年9月. 市川祐輔, 小西善二郎, 二村良彦.</p>
講演	<p>数式処理システム Mathematica 上における再帰除去システム. 情報科学技術フォーラム一般講演論文集(FIT2005), 第一分冊, pp.53-54, 2005年9月. 市川祐輔, 二村良彦, 上田和紀.</p> <p>単一後継関数を持つ再帰プログラムに対する再帰除去法の実装. 日本ソフトウェア学会大会論文集, Vol. 21, pp.396-400, 2004年9月. 市川祐輔, 小西善二郎, 二村良彦.</p> <p>Cumulative Method: Recursion Removal from Recursive Programs with One Descent Function. Workshop on Robust Software Construction (WRSC 2003), p.7, February 2003. Yusuke Ichikawa, Zenjiro Konishi, Yoshihiko Futamura.</p> <p>Recursion Removal from Recursive Programs with One Descent Function. Asian Symposium on Programming Languages and Systems (APLAS), China, pp.383-395, December 2002. Yusuke Ichikawa, Zenjiro Konishi, Yoshihiko Futamura.</p> <p>再帰除去におけるタプリング法の限界と累積関数法. 日本ソフトウェア学会大会第19回大会論文集, 6F-2. 2002年9月. 市川祐輔, 小西善二郎, 二村良彦.</p> <p>プログラム変換における数式処理の役割. 日本数式処理学会第11回大会, 2002年9月. 市川祐輔, 小西善二郎, 二村良彦.</p>

研 究 業 績

種 類 別	題名、 発表・発行掲載誌名、 発表・発行年月、 連名者（申請者含む）
講演	<p>単一後継関数を持つ再帰プログラムからの再帰除去. 情報処理学会プログラミング研究会, Vol. 43, No. SIG 8 (PRO 15), 論文誌プログラミング, p.126, 2002年9月. 市川祐輔, 松谷将寛, 小西善二郎, 二村良彦.</p> <p>整列法性能評価システムとランダムデータサーバーのデモ. 日本ソフトウェア科学会第18回大会ポスターデモセッション, P-1, 2001年9月. 伍偉鴻, 市川祐輔, 二村良彦.</p> <p>大規模リスト処理におけるGCの挙動解析の一例. 情報処理学会システム評価研究グループ, 2000年10月. 市川祐輔, 川田洋平, 箕一彦, 小西善二郎, 二村良彦.</p>

研 究 業 績

種 類 別	題名、 発表・発行掲載誌名、 発表・発行年月、 連名者（申請者含む）