

## 科学研究費助成事業（科学研究費補助金）研究成果報告書

平成25年5月15日現在

機関番号：11301

研究種目：基盤研究（C）

研究期間：2010～2012

課題番号：22500023

研究課題名（和文） 証明論に基づくコンパイラの系統的な構築法の研究

研究課題名（英文） A Study on Proof-Theoretical Foundations for Compiler Construction

研究代表者

大堀 淳 (OHORI ATSUSHI)

東北大学・電気通信研究所・教授

研究者番号：60252532

研究成果の概要（和文）：

本研究では、プログラミング言語のコンパイルの過程に現れる種々の中間言語は、直感主義的論理学の証明システムとして表現でき、それらの言語間の変換は、証明変換として表現できるはずである、との基本的な洞察を基礎とし、関数型言語のコンパイル過程を自然演繹システムから、コード言語を表現するある種のシーケント計算系にいたる証明変換の合成として表現でき、その変換可能性を示すメタレベルの証明から、コンパイルアルゴリズムが抽出できることを示した。

研究成果の概要（英文）：

Based on the novel observations that each of compiler intermediate languages can be represented as a proof system of the intuitionistic propositional logic, and that transformation between these languages corresponds to proof transformation, this research has shown that a compilation process of a functional language is represented by the composition of proof transformations from the natural deduction proof system to a variant of a sequent calculus that represents a code language, and that a compilation algorithm is mechanically extracted from the meta-level proof of the existence of such a proof transformation.

交付決定額

（金額単位：円）

	直接経費	間接経費	合計
2010年度	900,000	270,000	1,170,000
2011年度	1,100,000	330,000	1,430,000
2012年度	900,000	270,000	1,170,000
年度			
年度			
総計	2,900,000	870,000	3,770,000

研究分野：情報科学

科研費の分科・細目：ソフトウェア

キーワード：コンパイラ、証明論、最適化、プログラミング言語処理系

## 1. 研究開始当初の背景

今日の計算機システムでは、すべてのプログラムは機械語に翻訳され、フォン・ノイマン型逐次機械によって実行される。この事実は、すべてのプログラムの正しさは、プ

ログラムを機械語コードに翻訳するコンパイラの正しさに依存していることを意味している。この重要な位置づけから、正しいコンパイラ構築の試みは、計算機科学の発展とともに探求されてきた最重要な課題の一つ

である。コンパイラ研究の長い歴史の中で、型の解析や意味解析、コード生成、レジスタ割付け等の数々の研究がなされ、より正しいコンパイラの構築技術が蓄積されている、しかし残念ながら、正しさが保証されたコンパイラが構築できる技術が確立されたと言える状況には至っていない。

コンパイラもプログラムの一つであり、正しさが保証されたコンパイルの構築は、正しさが保証されたプログラム構築の一種である。この洞察から、従来のコンパイラの正しさの研究では、コンパイラを分析対象のプログラムと捕らえ、その性質を記述するメタ論理を構築し、プログラムとしてのコンパイラの正しさを証明する手法が試みられている。例えば代表的な成果に、Leroy による CompCert コンパイラなどがある。これらの成果から、このメタレベルの検証のアプローチも、十分に人的なリソースを注げば、将来、正しさが検証された実用コンパイラ構築の可能性を開くと期待される。しかしながら、このメタレベルのアプローチでは、検証対象となるコンパイラの間接言語やそれら进行操作するアルゴリズムと、それらアルゴリズムが正しくコンパイルするという性質を、検証システムが理解する複雑なメタ言語で表現する必要がある。この表現の過程およびその結果得られる正しさの概念そのものも高度に専門的であり、コンパイラの種々のアルゴリズムを開発しコードを書くプログラマにとって直感的に納得できるものからは距離があるという欠点がある。

本研究が提案するアプローチは、コンパイラが言語の翻訳システムであるとの洞察を基礎とし、コンパイラを、あらかじめ確立された正しい翻訳の方式に従って構成する、というものである。従って、もしこのようなアプローチが確立されるならば、この方式で構築されたアルゴリズムは、その構成上正しいと言える。しかしこのようなアプローチは、報告者の知る限り殆ど試みられていない。本研究では、直感主義的論理学を、言語翻訳システムを定義する枠組みとし、Curry-Howard 同型関係の概念を拡張することによってこの目的を達成することを試みる。プログラミング言語と直感主義的証明論の関係は、特に Curry-Howard 同型関係の概念は、1970 年代から広く知られ、プログラミング言語の設計などの基礎としての役割を果たしてきた。しかしながら、この関係を、コンパイラの系統的な設計や実装、さらにその正しさの証明等に使用出来るとの洞察やその試みは殆ど行われていなかった..

## 2. 研究の目的

本研究の一般的かつ最終的な目的は、ソース言語から多数の中間言語を経てコード

言語へと変換される高水準言語のコンパイルの全過程の証明論的解釈を確立し、その証明論的解釈に基づき、正しい、すなわち型と操作的意味を保存する、コンパイルアルゴリズムを系統的に抽出する方式を確立することである。これは、正しさが保証されたコンパイラの構築を可能とする新しいアプローチである。従来の、論理学をメタレベルの検証に用いるアプローチと異なり、アルゴリズムの構成構造そのものからそのアルゴリズムの正しさが保証されるため、正しいコンパイラのより系統的な構築が可能となると期待される。

本研究が基礎とする直感主義的証明論に基づくコンパイル課程の系統的解釈の確立は、その枠組みやさらにその着想さえ殆ど検討されていなかった独創性の高いものであり、また、正しいコンパイラの構築方法の確立は、前述の通り、計算機科学において長く研究されてきた困難な課題である。そのような性質上、本研究が最終的に目指す証明論に基づく正しいコンパイラの系統的な導出は、計算機科学の本科学研究費補助金の交付期間内に完結するとは期待できない。本研究では、科学研究費補助金の基盤研究の趣旨に従い、本研究が提案する証明論的アプローチに関する基礎的・基盤的な研究を行い、この新規的かつ挑戦的な課題の達成への道を開くことを目指す。

本研究の具体的な目標は、ラムダ計算から、A 正規形を経て機械語コードを生成するコンパイルの過程を、

- (1) 自然演繹システムをシーケント計算系に変換する過程、
- (2) シーケント計算の Commutative Conversion に関する正規形に変換する過程、
- (3) シーケント計算系を逐次シーケント計算系に変換する過程、さらに、
- (4) 制限された逐次シーケント計算系に変換する過程

の組み合わせで表現し、それら各段階の系統的なアルゴリズムの構築の基礎を与えることである。以上に加え、本研究ではさらに、以上のようなコンパイラの証明論的な解釈のアプローチに基づき、コンパイラの種々の機能や種々のコンパイル方式に関するより系統的な知見を得ることを目指す。

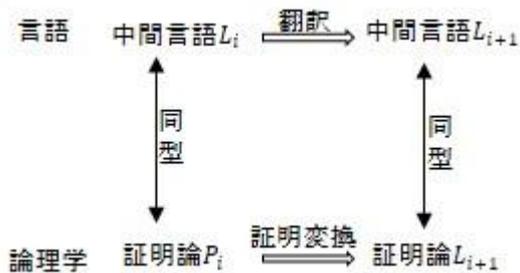
## 3. 研究の方法

前述の通り、本研究が目指すアプローチは、コンパイルアルゴリズムそのものの構築の枠組みを証明論的に再構成し、その構成法に従いアルゴリズムの正しさそのものを確立しようとする試みである。その基礎とする先行研究結果は、報告者によるコード言語の証明論的基礎 (Ohori, A Proof Theory for

Machine Code, ACM TOPLAS 2007), バイトコードの検証論 (Higuchi and Ohori, A Static Type System for Javabytecode, ACM TOPLAS 2006), 証明論に基づくレジスタ割付方式 (Ohori, Register Allocation by Proof Transformation, J. Sci. Prog. 2004), および A-normal の証明論 (Ohori, A Curry-Howard Isomorphism for Compilation and Program Execution, TLCA 1999) の一連の研究である。

これら一連のプログラミング言語の証明論的探求の中で特に、コード言語の証明論的基礎 (TOPLAS 2007) において、従来低レベルのアドホックな体系とみなされていたコード言語が、直感主義的論理学の証明システムとして表現できることが示された。具体的には、機械語コードの型の正しさの検証システムが逐次シーケント計算と呼ばれる証明システムの証明可能性に対応し、コードの実行ステップが証明システムにおける証明のカット除去プロセスに対応することが厳密に示されている。

本研究では、この研究によって示されたコード言語と証明論の同型関係とそれを導くために新たに導入された証明論的概念を、コンパイルの途中に現れる全ての中間言語 ( $L_1$  から  $L_n$ ) にも導入し、各中間言語について、その言語と同型の証明システム ( $P_1$  から  $P_n$ ) を厳密に定義し、さらに、それら証明システムが同値であること、すなわち、 $P_i$  と  $P_{i+1}$  において、 $P_i$  の任意の証明は  $P_{i+1}$  でも証明できることを証明し、その (メタレベルの) 証明から、対応する  $L_i$  のプログラムが  $L_{i+1}$  のプログラムに変換できることを示す戦略をとる。この関連を以下の図に示す。



#### 4. 研究成果

本研究では、コンパイルのソース言語を、型付き高水準関数型言語とし、ターゲット言語をレジスタまたはスタックを使って動く抽象的なコード言語とした。実際のコンパイルでは、ソース言語からターゲットコード言語に至る変換過程で、多数の中間言語が定義され多段階の変換が行われる。例えば、我々の究極的な目標である正しさが保証されたコンパイラの対象として想定している、我々が開発している次世代関数型言語

SML#言語では、20 段階以上のコンパイルステップをへて機械語コードに変換されるが、コンパイルの証明論的基礎の確立を目指す最初の段階である本研究では、中間言語として A-正規形のみを考え、コンパイルを、ソース言語から A-正規形を経てコード言語に変換される過程と捉えた。

本研究の第一の目標は、コンパイル過程に現れる各中間言語に直接対応する直感主義的論理学の証明論を定義することである。1970 年代からよく知られている通り、ソース言語は自然演繹証明システムに対応する。また、報告者の先行研究によって A-正規形は、Kleene によって提案された  $G_3$  と呼ばれるシーケント計算系に対応することが示されている。さらに、報告者の先行研究によって、コード言語に対する逐次シーケント計算と呼ばれる直感主義的な証明システムが定義されている。これらの中で、逐次シーケント計算は、コードの実行に直接対応するように定義された証明システムであるため、証明システムにおけるシーケントの導出構造がコードの実行環境と同型になるように定義され、それを基礎にカット除去プロセスとコードの操作的意味論との対応が確立されている。しかしながら、従来の自然演繹システムと Kleene スタイルシーケント計算系の定義は、ラムダ計算および A-正規形の実行環境との対応がとれておらず、コンパイルの基礎としては不適合である。そこで本研究ではまず、これら 2 つの証明システムに対して、言語の実行環境に対応する「値証明」と「環境証明」の概念を導入して拡張した。これによって、関数型言語のソース言語としてのラムダ計算と A-正規形と同型の証明システムを得ることができた。

上記のように洗練された自然演繹システムと Kleene スタイルシーケント計算に対して、先行研究にて示した逐次シーケント計算にカット除去定理の構造にならば、それぞれの操作的意味論と同型のカット除去定理を確立した。さらに、これら 3 つの証明システム間の証明変換可能性に関するメタ定理を証明した。このメタ定理から、それぞれの計算系と言語との同型関係を用いて、プログラムの変換アルゴリズムが導出できることが確認された。理論を完成させるためには、さらなるいくつかの課題の解決が必要であるが、これら内容により、当初の目標であるコンパイルの証明論的基礎付けの道筋ができたと言える。その骨格は、内容を検証するとともに大学院の先端講義ノートとして作成し、講義にて使用している。現在、これらを網羅した 3 部からなる論文を準備中である。

以上の中核的な成果に加え、証明論を様相論理によって洗練することにより、ゲームプログラ等の高度なプログラムの系統的な生

成等, コンパイラの種々の機能の証明論的な基礎づけの可能性に関する洞察を得た.

#### 5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文] (計2件)

- ① 篠埜 功, 大堀 淳. SML#へのC言語の埋め込み. コンピュータ ソフトウェア, Vol. 29 (2) pp. 193-203, 2012. 査読有. (DOI: 10.11309/jssst.29.2\_193)
- ② 上野雄大, 大堀淳. 多相レコード計算に基づく軽量の第一級オーバーロードの設計と実装, コンピュータソフトウェア, 29(1), pp. 191-210, 2012. 査読有. (DOI: 10.11309/jssst.29.1\_191)

[学会発表] (計8件)

- ① 藤井貴啓, 上野雄大, 森畑明昌, 大堀淳. SML#のデータベース連携機能を活用したウェブアプリケーション構築技術. 第15回プログラミングおよびプログラミング言語ワークショップ, 2013年3月5日、福島県会津若松市.
- ② 深澤 優鷹, 上野 雄大, 森畑 明昌, 大堀 淳. Ruby の操作的意味論の形式的定義に向けて. 第15回プログラミングおよびプログラミング言語ワークショップ(ポスターセッション), 2013年3月5日、福島県会津若松市.
- ③ 斎藤 皓, 上野 雄大, 森畑 明昌, 大堀 淳. SML#のSQL 統合への group by の導入. 第15回プログラミングおよびプログラミング言語ワークショップ (ポスターセッション), 2013年3月5日、福島県会津若松市.
- ④ 小石真人, 森畑明昌, 大堀淳. 二次元最大重みと問題のプログラム変換に基づく解法. 第15回プログラミングおよびプログラミング言語ワークショップ, 2013年3月4日、福島県会津若松市.
- ⑤ Atsushi Ohori, Katsuhiko Ueno. Making standard ML a practical database programming language. Proceedings of ACM International Conference of Functional Programming, pages 307-319, September 21, 2011, Tokyo, Japan. (DOI:10.1145/2034773.2034815)
- ⑥ Atsushi Ohori. Development of SML# - making ML an ordinary practical language (invited talk). ACM ML Workshop, September 18, 2011, Tokyo, Japan.
- ⑦ 松島勇介, 上野雄大, 森畑明昌, 大堀淳. 宣言的記述からの関数型言語によるゲ

ームプログラムの導出. 第13回プログラミングおよびプログラミング言語ワークショップ, 2011年3月10日、北海道札幌市. (本発表は、学生の部発表賞を受賞)

- ⑧ 高橋和将, 森畑明昌, 上野雄大, 大堀淳. 生存区間を変数名とする中間表現の実装と、それに基づく最適化, 第13回プログラミングおよびプログラミング言語ワークショップ, 2011年3月9日、北海道札幌市.
- [図書] (計0件)

[産業財産権]

○出願状況 (計0件)

名称:  
発明者:  
権利者:  
種類:  
番号:  
出願年月日:  
国内外の別:

○取得状況 (計0件)

名称:  
発明者:  
権利者:  
種類:  
番号:  
取得年月日:  
国内外の別:

[その他]

ホームページ等

#### 6. 研究組織

##### (1) 研究代表者

大堀 淳 (OHORI ATSUSHI)  
東北大学・電気通信研究所・教授  
研究者番号: 60252532

##### (2) 研究分担者

上野 雄大 (UENO KATSUHIRO)  
東北大学・電気通信研究所・助教  
研究者番号: 60551554

森畑 明昌 (MORIHATA AKIMASA)  
東北大学・電気通信研究所・助教  
研究者番号: 10582257