

## 審査結果の要旨

論文提出者氏名 千代英一郎

プログラムの実行時の振る舞いに関する性質を静的に調べるプログラム解析器は、コンパイラが生成する実行コードの性能を左右する主要な構成要素である。その不具合はコンパイラが生成する実行コードの不具合につながるため、その精度や効率とともに高い信頼性が求められている。信頼性の高いプログラム解析器を開発する有力な方法のひとつに対象言語の形式的意味定義に基づく方法が存在するが、C 言語に対しては、このような開発は未だ行われるに至っていない。その最大の問題点は、プログラム解析器の開発の基盤となりうる適切な形式的意味定義が存在しない点にある。

本研究の主な貢献は、C に対する形式的意味定義に基づくプログラム解析器の開発を実現したことである。ここでは、形式的意味定義に適した中間表現を提案し、それが実用的な C プログラム解析器の開発に有用であることを示している。また、既存のコンパイラで用いる既存の中間表現の仕様記述法を提案し、コンパイラと整合性のある解析器の構成手法を与えている。

本論文は、**A Logic-Based Approach to Developing Analysers for C Programs** (論理にもとづく C プログラム解析器の開発に関する研究) と題し、5 部より構成され、主要部は第 II 部から第 IV 部である。第 II 部と第 III 部では、高い信頼性を持つ C プログラム解析器の開発基盤となる中間表現 **RML (Register Memory Language)** を導入し、それに基づく信頼性の高いポインタ解析器の開発法を扱っている。第 IV 部では、中間表現の仕様記述に適したデータモデル **IIR (Identifier-based Intermediate Representation data model)** を導入し、それをを用いて既存の C コンパイラの中間表現の仕様を記述している。これら各部で中心的な役割を果たす **RML** と **IIR** の関係は相補的であり、**RML** が C プログラムの抽象的な論理式表現であるのに対して、**IIR** は **RML** の具体的表現を定義する際の仕様記述方法を与えるものであり、本論文ではその関連を明確に述べている。

第 I 部 **Introduction** は導入部である。

第 1 章 **Introduction** では、本研究の対象、動機、目的、方法を明らかにし、

関連する従来研究の問題点を指摘している。

第 2 章 Preliminaries では、本論文で用いる基本的な用語、表記法を定義している。

第 II 部 Definition of RML では、中間表現 RML の定義を行っている。

第 3 章 The C Standard では、C の標準言語規格を分析し、形式的意味を定義する際に生じる問題点を明確にしている。

第 4 章 Overview of RML では、以下の章で扱う RML の形式的意味定義の概要を示している。

第 5 章 Static Semantics および 第 6 章 Dynamic Semantics では、それぞれ RML の静的、および動的な形式的意味の定義を与えている。

第 7 章 Discussion and Related Work は第 II 部のまとめとして、関連研究との比較を行い、その利点・欠点を議論している。

第 III 部 Systematic Development of a Pointer Analyser では、第 II 部の成果を C プログラムのポインタ解析器の開発に適用して得られた結果を示している。

第 8 章 Pointer Analysis Problem では、ポインタ解析の目的、関連研究、研究課題を述べている。また、第 II 部で与えた形式的意味記述を用いて、ポインタ解析問題を形式的に定義している。

第 9 章 Abstract Semantics では、第 8 章で与えた問題定義に基づき、形式的意味をその構造に沿って抽象化した抽象的意味を定義している。

第 10 章 Algorithm では、第 9 章で得られた抽象的意味から解析アルゴリズムを導出する方法を示している。

第 11 章 Concluding Remarks では第 III 部をまとめて、ポインタ解析問題への本手法の適用の優位性を論じている。

第 IV 部 Method for Specifying Static Semantics of IR では、プログラムの中間表現(Intermediate Representation)の仕様記述を支援する手法を扱っている。

第 12 章 Introduction では、プログラムの中間表現に関する問題点、およびここで提案する手法の概要を示している。

第 13 章 IIR では中間表現データモデル IIR を定義している。

第 14 章 Case Study では、提案手法の適用事例として、製品 C コンパイラで用いている中間表現の仕様を記述した結果を示している。

第 15 章 Other Applications では、IIR のさらなる応用可能性を示し、プロ

グラムを関係の集合としてモデル化する IIR の特徴が仕様記述以外にも、論理型言語を用いた宣言的なプログラム解析に有用であることを主張している。

第 16 章 Discussion と第 17 章 Related Work では、提案手法を関連研究と比較してその利点・欠点を議論している。

第 18 章は第 IV 部のまとめとして、IIR の有効性を述べている。

第 V 部 Conclusion は本論文の結論部である。

第 19 章 Conclusions and Future Works では、本研究の成果をまとめ、今後の課題について述べている。

本論文において提案した C の形式的意味定義に基づくプログラム解析器の開発例であるポインタ指示先解析器の設計とその正しさの証明は、型変換と共用体を扱う解析器に関する初めてのものである。また、既存のコンパイラで用いるプログラム解析器を開発する上で障害となっている中間表現の仕様記述方法を提案して、既存の C コンパイラの中間表現の仕様を形式的に記述できることを示した成果は、実用的なプログラム解析器の開発に有用な視点を与えるものである。

以上のように、本論文はこれまで十分ではなかった C 言語に対する形式的意味記述に基づくプログラム解析器の開発手法を提案してその有効性を例示するとともに、既存のコンパイラで用いられている中間表現の仕様記述法を与えて実用的な解析器の構成法を実証的に明らかにしたものとして、この分野に少なくない貢献を果している。すなわち、本研究は情報理工学に関する研究的意義とともに、情報理工学における創造的実践に関し価値が認められる。よって本論文は博士（情報理工学）の学位請求論文として合格と認められる。