様式C−19

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 6 月 5 日現在

機関番号：14301
研究種目：基盤研究 (B)
研究期間：2010　～　2012
課題番号：22300008

研究課題名（和文）　バグのないソフトウェア構築環境に関する研究の新展開

研究課題名（英文）　New development of research on bug-free software construction environment

研究代表者
佐藤 雅彦　（SATO MASAHIKO）
京都大学・大学院情報学研究科・名誉教授
研究者番号：20027387

研究成果の概要（和文）：本研究では，ソフトウェアの安全性を保障するための方法論として，バグのないソフトウェアを構築する環境を実現することの重要性を指摘し，具体的なシステムのプロトタイプを実装した．とくに，計算と論理を融合した，新しいプログラミング言語を設計・実装し，その言語を用いてシステムを実装した．このため，実装したシステム自体の安全性についても，原理的に検証可能である．システムとユーザとのインターフェースもこの言語を用いて実装した．

研究成果の概要（英文）：　We started our research by setting the goal of realizing a computer environment for developing bug-free softwares, so that we can guarantee the safety of computer software. We have achieved the goal by implementing a prototype system. In particular, as a part of the system, we implemented a programming language which can be used to reason about the properties of the system itself. The interface between the user and the system is also implemented by this programming language.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
2010 年度	5,700,000	1,710,000	7,410,000
2011 年度	4,700,000	1,410,000	6,110,000
2012 年度	3,400,,000	1,020,000	4,420,000
総　計	13,800,000	4,140,000	17,940,000

研究分野：総合領域
科研費の分科・細目：情報学、ソフトウェア
キーワード：ソフトウェア検証、クラス理論、ソフトウェアの安全性、型理論

1.研究開始当初の背景

(1) ソフトウェアの安全性を検証するための理論として，論理学の手法を用いて形式的体系の性質の記述及び検証を数学的に厳密な方法で行なう言語体系であるロジカル・フレームワーク（論理枠組）が提案されている．形式的な計算体系であるプログラミング言語をロジカル・フレームワーク上で記述することにより，ソフトウェアに求められる仕様を正確に記述し，与えられたソフトウェアがその仕様を満足することを厳密に検証することが可能となる．さらにロジカル・フレームワークを計算機上に実装することができれば，これらの記述・検証の機械的に正確な検査が可能となり，ソフトウェアの品質面，安全面の信頼性が高まることが期待される．

(2) このことから，ソフトウェアの安全性を保証するためには

① 形式的体系（ここではプログラミング言語，およびソフトウェア）を正確に記述するためのロジカル・フレームワークの理論研究，

② ロジカル・フレームワーク上に記述された推論過程が正しいかどうかを機械的・形式的に検査する技術の研究

といった，理論的基盤を与えることが重要である．さらに，バグのないソフトウェアを効率良く開発するためには，これらの理論的基盤をとりこんだソフトウェア構築環境を実現することが非常に重要である．ここで注意すべきは，ソフトウェア自身だけではなく，ソフトウェアの性質を記述するための理論を含めた階層をまるごと対象とする必要がある，という点である．以上の目的の実現のためには，計算の概念を利用したより自然な形での証明の構築を可能とする計算と論理を融合したロジカル・フレームワークを設計する必要がある．

2.研究の目的

本研究は，議論対象の階層を取り扱うためのメタ変数の概念を取り入れ，それを用いて計算と論理を融合したロジカル・フレームワークの理論構築とその実装を目的とする．より具体的には以下を目的とする．

(1) メタ変数の形式化．メタ変数とは「プログラム」「論理式」といった形式的体系を構成する要素を表す変数で，形式的体系の定義における一般性の高い表現を実現するために不可欠な概念である．上で述べたように，ソフトウェア自身だけでなく，そのソフトウェアに関してメタな議論を行うための論理的体系をも形式的に記述するために，通常非形式的に扱われるメタ変数を形式的な概念として取り扱うための理論を構築する．

(2) ロジカル・フレームワークの基盤となる形式言語体系の研究．上で述べたメタ変数の理論を元に，ソフトウェアという計算体系とその理論である論理体系の両方を統一的に記述できるような共通の言語を開発する．この目標は，ある種の計算体系と論理体系が本質的に同一であるというカリー・ハワード同型対応に基づくと達成可能であると予想できる．これにより，計算と論理を融合したロジカル・フレームワーク実現の基盤を確立する．

(3) ロジカル・フレームワーク実装用プログラム言語の設計と実現．(1)，(2) で述べたメタ変数の概念を取り入れたロジカル・フレームワークを計算機上に実現するためには，既存のプログラム言語で適当なものは見当らない．そのため，本研究では，メタ変数およびメタ変数が表すところの言語対象を自然に扱えるようなプログラム言語を設計し実装する．

(4) ロジカル・フレームワークをとりこんだソフ

トウェア構築環境の実装. 以上の理論的･技術的研究を総合し, ロジカル･フレームワークを基礎としたソフトウェア構築環境を提案する. この環境では, ソフトウェア開発と同じ枠組で, 上記の検証技術,証明支援技術を利用できるため , バグのないソフトウェアを効率良く開発できる.

3.研究の方法

(1) 計算と論理を融合した自然枠組の設計. 我々がこれまでに提案した形式的記述体系である自然枠組 (Natural Framework, NF) をもとに, 計算と論理が自然に融合するように拡張を行う. まず, そのために NF の構文的対象を記述するための式の理論を再設計する.この式の理論により, 本研究で構築するプログラム言語が扱う対象がすべて式により再現されるようになる. さらに対象はすべて固有のクラスのインスタンスとして実現され,クラスも言語で扱うことのできる対象となる. この結果, プログラム等の数学的対象を記述する式と命題を記述する式がそれぞれ個別のクラスに属するようになり, プログラムと命題を構文的に区別できようになる.

(2) メタ変数の概念の形式化. メタ変数に関する理論的な考察を行う. 非形式的には既に幅広く用いられているメタ変数の概念を, 理論的に考察し, できる限りその意味を正確に反映する形で形式化する. 本研究では, 型理論の手法を用いて, メタ変数を扱うための形式的計算体系を構成する方法をとるが, この際, メタ変数の概念を, 既存の様々な計算体系に付加することができるような形で定式化できる方法で構成する. この手法の正しさと,汎用性を確認するために, 型なし λ 計算や単純型付 λ 計算などにこの方法を適用してメタ変数の概念を追加し, この体系に関して, 合流性や停止性などの期待される性質を証明する.

(3) 変数束縛を持つデータ構造に関する性質の証明技法の研究. プログラムにおけるパラメータを持つ手続き, 全称化を伴う論理式といった概念を扱うためには, 変数束縛のある構造を対象として議論を行う必要がある. しかし, このような構造に関しては, 素朴な帰納法による証明技法は正しくないことが知られている. そのため, 変数束縛を持つデータ構造に関する性質を証明するための技法を研究し, NF 上の証明検査に適用する.

(4) 自然枠組の実装. NF に基づく処理系として既に実装されている教育用システムである CALをもとにして, (1) で基礎設計した自然枠組および,証明検査アルゴリズムのプロトタイプを実装し,性能評価をする.

4.研究成果

(1) ソフトウェア構築環境 NF 実現のためのプログラミング言語 Ez の実装.研究の初期段階で, NF 実現のためには, それを実装するプログラミング言語を設計･実装することの必要性が認識された. この言語は, NF で扱う基本的なデータである命題や証明を自然に扱えるように設計し, 言語 Ez として実装することができた.

(2) 変数束縛の理論については, その基礎として古典的な λ 計算における束縛機構について, これまで知られていなかった新しい実現の方式を提案し, その性質を厳密に証明した. より具体的には, 従来の方式では, 自由変数をそのまま束縛変数として利用するために, 代入操作の定義が複雑になっていた. 新しい方式では, 自由変数の束縛範囲内での出現の分布を考慮して, 変数の高さを計算する関数を定義し, その高さを用いて束縛変数を定義することにより, 代入操作が自然に定義できることを証明した.

(3) ロジカル･フレームワーク NF は Ez を用いてプロトタイプを実装し, 教育用システム

CALで必要とされる機能を実現することができた．また,この過程において，クラス理論に基づく NF が型理論に基づくロジカル・フレームワークとくらべて，多様な論理体系を柔軟に実現できることを確認することができた．

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　4　件）

(1) Randy Pollack, Masahiko Sato and Wilmer Ricciotti, A Canonical Local Representation of Binding, Journal of Automated Reasoning, Vol. 49, pp. 185-207, 2012. 査読有

(2) Kensuke Kojima and Atsushi Igarashi, Constructive linear-time temporal logic: Proof systems and Kripke semantics, Information and Computation, Vol. 209(12), pp. 1491--1503, 2011. 査読有

(3) Masahiko Sato, and Randy Pollack, External and internal syntax of the lambda-calculus, Journal of Symbolic Computation, Vol. 45, pp. 598--616, 2010. 査読有

(4) Takeshi Tsukada and Atsushi Igarashi, A logical foundation for environment classifiers, Logical Methods in Computer Science, Vol. 6(4:8), pp. 1-43, 2010. 査読有

〔学会発表〕（計　8　件）

① 佐藤雅彦，Viewing lambda-terms through maps, 3rd Workshop on Proof Theory and Rewriting, 2013 年 3 月 7 日，石川県立美術館（金沢市）

② 山本章博,整数計画ソルバを用いた帰納論理プログラミング,人工知能学会 第 89 回人工知能基本問題研究会(SIG-FPAI)，2013 年 01 月 31 日～2013 年 02 月 01 日,岩手県立大学

③ 佐藤雅彦，Essence of de Bruijn Index, 37th TRS meeting, 2012 年 11 月 7 日，岩沼屋（仙台市）

6.研究組織

(1) 研究代表者

佐藤 雅彦　(SATO MASAHIKO)
京都大学・大学院情報学研究科・名誉教授
研究者番号：　20027387

(2) 研究分担者

湯淺 太一　(YUASA TAIICHI)
京都大学・大学院情報学研究科・名誉教授
研究者番号：　60158326

山本 章博　(YAMAMOTO AKIHIRO)
京都大学・大学院情報学研究科・教授
研究者番号：　30230535

五十嵐 淳　(IGARASHI ATSUSHI)
京都大学・大学院情報学研究科・教授
研究者番号：　40323456

中澤 巧爾　(NAKAZAWA KOJI)
京都大学・大学院情報学研究科・助教
研究者番号：　80362581