様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成　２９　年　　６　月　２３　日現在

機関番号：　１４３０１
研究種目：　基盤研究(B)（一般）
研究期間：　2013～2016
課題番号：　２５２８００２５

研究課題名（和文）クラス理論に基づく自己拡張可能なソフトウェア検証体系の構築

研究課題名（英文）Construction of a self-extendable software verification system based on class theory

研究代表者
佐藤　雅彦（Sato, Masahiko）

京都大学・情報学研究科・名誉教授

研究者番号：２００２７３８７

交付決定額（研究期間全体）：（直接経費）　　13,500,000円

研究成果の概要（和文）：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う言語体系である論理枠組が提案されている本研究では，従来の論理枠組は型理論の上に構築されているのに対してクラス理論の上に構築することを実現した．

ここでのクラス理論は本研究で新しく提案した理論であり，型理論にない柔軟性があり，さらにクラス全体のクラスを矛盾なく扱えるという特徴がある．この特徴のため，論理枠組の中で，枠組自身に言及し，その構造を解析することができた．とくに重要な成果として，変数の束縛機構を分析し，α同値性の概念について新しい見方での定義を与えることができた．

研究成果の概要（英文）：In order to verify the safety of software, the concept of Logical Framework is proposed. A logical framework can be used to internally realize formal systems and prove their properties in a rigorous manner. In contrast to the fact that most logical frameworks are based on type theory, we realized our logical framework based on class theory.

We proposed the new class theory in this research. The class theory is more flexible than type theories and enjoys the property that it can deal with the class of all classed in a consistent way. Due to this property, our logical framework can refer to the framework itself and can analyze its structure by itself. We analyzed the mechanism of variable binding in the lambda calculus, and could give a new definition of the notion of the alpha-equivalence.

研究分野：　ソフトウェアの基礎理論

キーワード：　クラス理論　型理論　証明検証　ソフトウェアの安全性

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

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) が提案されてきていた．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た．

(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，および，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，ソフトウェア理論自体の正しさをロジカル・フレームワークの上で議論するためには，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も含めた階層 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，自己拡張可能なロジカル・フレームワークを設計する必要があった．

2. 研究の目的

(1) 近年のインターネット／計算機の爆発的な普及，それに伴う社会基盤としての計算機の重要性の増加により，ソフトウェアの安全性など，ソフトウェアの品質に対する要求はますます高まっている．しかし，実際のソフトウェア開発においては，ソフトウェア構築環境とは別に検証システムを用意しなければならないなど，安全性の面で充分とは言えず，実際，バグを含んだソフトウェアが重要な箇所で利用され，重大な障害が発生している例も多い．

(2) 本研究は，この要求に応えるため，バグのないソフトウェアを構築するためのクラス理論に基づく理論的基盤を与え，さらにそれを用いて (ユーザによる自己拡張を許す) 自然枠組 (Natural Framework) とよばれるソフトウェア実行環境およびソフトウェア検証環境を同時に提供するシステムを計算機上に実装することを目的とした．

3. 研究の方法

(1) 基礎理論研究．

本研究で構築するシステムがメタレベルと対象レベルをシステム自身で記述，操作でき，さらに自己拡張可能になるようにするために必要な概念であるクラス理論，自然枠組およびメタ変数を中心にした理論研究を行った．

本研究で提案したクラス理論は，既存のロジカル・フレームワークの多くがその基礎としている型理論の弱点を取り除き，さらにより単純な構成原理のもとに，自己拡張可能性等の，型理論にない柔軟性を持つように設計することを考えた．とくに次のことに着目して研究を進めた．型理論においては，一旦理論が固定されると，その型理論でユーザが新しく構築できる型は，その理論で許されるスキーマにあてはまるものに限定される．このため，ユーザが本当にほしい型を自然に実現することはできず，既存の型を組合せることで不自然な形で実現することになり，ユーザによる自己拡張の自由はない．提案したクラス理論では，ユーザがほしい対象のクラスを，そのクラスの対象が満足すべき「性質」から直接，ユーザ自身が構築できる仕組みを提供するこ

とができた.

(2) システム実装.
自然枠組システム実装のために用いる言語として，自然枠組記述言語自身を用いる予定であるので，研究の前半ではこの言語の実装を行い，後半ではこの言語を用いて自然枠組システムの実装を行った．実装にあたっては Scheme の処理系のひとつである Racket を用いた.

4. 研究成果
(1) 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なった．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した．本研究では，今回提案するクラス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ったが，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した．この手法の正しさと，汎用性を確認するため，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ことができた.

(2) 変数束縛を持つデータ構造に関する研究.
プログラムにおけるパラメータを持つ手続き，全称化 (「すべての〜について」) を伴な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なう必要がある．しかし，このような構造に対しては，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，クラス理論の立場から研究し，自然枠組上の証明検査に適用した．この成果は，本研究代表者が変数束縛機構に関して共同研究を行っている米国ハーバード大学の Randy Pollack 博士およびドイツミュンヘン大学の Helmut Schwichtenberg 教授との共同研究によるものである ［①］

(3) 研究成果発表.
本研究の成果の一部は，ドイツ Fishbachau における EU 諸国の大学院生，研究者を対象とする「秋の学校」での連続講義として発表し［②］，またカナダ Toronto における研究集会での招待講演として発表した［③］.

〈引用文献〉
① Masahiko Sato, Randy Pollack, Helmut Schwichtenberg, Takafumi Sakurai, Viewing lambda-terms through maps, Indagationes Mathematicae, 査読有, 24 巻, 2013, 1073 – 1104
② Autumn school ”Proof and Computation”, 3rd to 8th October 2016, Fishchbachau, Germany. http://www.mathematik.uni-muenchen.de/~schwicht/pc16.php
③ Semantic Representation of Mathematical Knowledge Workshop, February 3 – 5, 2016.

5. 主な発表論文等
［雑誌論文］(計 15 件)
① Masahiko Sato, Randy Pollack, Helmut Schwichtenberg, Takafumi Sakurai, Viewing lambda-terms through maps, Indagationes Mathematicae, 査読有, 24 巻, 2013, 1073 – 1104
DOI: 10.1016/j.indag.2013.08.003

② Kensuke Kojima and Atsushi Igarashi, A Hoare logic for SIMT programs, Lecture Notes in Computer Science, 査読有, 8301 巻, 2013, 58 – 73

③ 高橋翔大，青戸等人，外山芳人，ボトムアップ最内項書き換えシステムの最内到達可能性，コンピュータソフトウェア，査読有, 31 巻, 2014, 75 – 89

④ 佐藤洸一，菊地健太郎，青戸等人，外山芳人，項

書き換えシステムの変換を利用した帰納的定理自動証明，コンピュータソフトウェア，査読有, 32 巻, 2015, 179 – 193

⑤ 中嶋辰成，青戸等人，外山芳人，書き換え帰納法に基づく帰納的定理の決定可能性，コンピュータソフトウェア，査読有, 31 巻, 2014, 294 – 306

⑥ Takahito Aoto, Yoshihito Toyama and Kazumasa Uchida, Proving confluence of term rewriting systems via peristency and decreasing diagrams, Lecuter Notes in Computer Science, 査読有, 8560 巻, 2014, 46 – 60

⑦ K. Kikuchi, T. Sakurai, A Translation of Intersection Types for the lambda-mu-calculus, Lecuter Notes in Computer Science, 査読有, 8858 巻, 2014, 120 – 139

⑧ Takaki Suzuki, Kentaro Kikuchi, Takahito Aoto and Yoshihito Toyama, Critical pair analysis in nominal rewriting, EPiC Series in Computing, 査読有, 39 巻, 2016, 156 – 168

⑨ Koichi Sato, Kentaro Kikuchi, Takahito Aoto and Yoshihito Toyama, Correctness of context-moving transformations for term rewriting systems, Lecuter Notes in Computer Science, 査読有, 8858 巻, 2015, 331 – 345

⑩ Takaki Suzuki, Kentaro Kikuchi, Takahito Aoto and Yoshihito Toyama, Confluence of orthogonal nominal rewriting syterm revisited, Leibniz International Proceedings in Informatics, 査読有, 36 巻, 2015, 301 – 317

⑪ 佐藤雅彦，フレーゲ哲学の現代的意義 – 野本和幸著『フレーゲ哲学の全貌』を読む –，科学哲学，査読有, 49 巻, 2016, 67 – 84

⑫ Qi Tan, Kohei Suenaga, Atsushi Igarashi, An Extended Type System for Memory-Leak Freedom, 日本ソフトウェア科学会第 33 回大会論文集，査読無，2016

⑬ Vincent van Oostrom and Yoshihito Toyama, Normlisation by random descent, Leibniz International Proceedings in Informatics, 査読有, 52 巻, 2016, 32:1 – 32:18

⑭ Takahito Aoto and Yoshihito Toyama, Ground confluence prover based on rewriting induction, Leibniz International Proceedings in Informatics, 査読有, 52 巻, 2016, 33:1 – 33:12

⑮ Takaki Suzuki, Kentaro Kikuchi, Takahito Aoto and Yoshihito Toyama, Critical pair analysis in nominal rewriting, EPiC Series in Computing, 査読有, 39 巻, 2016, 156 – 168

［学会発表］(計 8 件)

① 鈴木貴樹，菊地健太郎，青戸等人，外山芳人，名目書き換えシステムの合流性について，第 16 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，2014 年 3 月 4 日～2014 年 3 月 7 日，熊本県阿蘇市阿蘇温泉「阿蘇の司ビラパークホテル」

② Masahiko Sato, A name-free lambda calculus, JAIST Logic Workshop Series 2015: Constuctive Computability (招待講演)，2015 年 3 月 2 日～2015 年 3 月 6 日，Shiinoki Cultural Complex, Kanazawa

③ 佐藤雅彦，計算と論理，情報処理学会第 77 回全国大会 (招待講演)，2015 年 3 月 17 日，京都大学

④ Masahiko Sato, A name-free lambda calculus, TPP 2014: Theorem proving and provers for reliable theory and implementations, 2014 年 12 月 3 日～2014 年 12 月 5 日，Kyushu University

⑤ Masahiko Sato, A name-free lambda calculus, JAIST Logic Workshop Series 2015: Constuctive Computability (招待講演)，2015 年 3 月 2 日〜2015 年 3 月 6 日，Shiinoki Cultural Complex, Kanazawa

⑥ 佐藤雅彦，私の基礎研究，日本ソフトウェア科学会第 32 回大会 (招待講演)，2015 年 9 月 10 日，早稲田大学理工学部

⑦ 小林恵，五十嵐淳，参照を備えた多段階計算のための多相的型システム，日本ソフトウェア科学会第 32 回大会，2015 年 9 月 11 日，早稲田大学理工学部

⑧ Qi Tan, Kohei Suenaga, Atsushi Igarashi, An Extended Behavioral Type System for Memory-Leak Freedom, 日本ソフトウェア科学会第 33 回大会，2016 年 9 月 6 日〜2016 年 9 月 9 日，東北大学東北大学電気通信研究所

6．研究組織

(1) 研究代表者

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

(2) 研究分担者

外山 芳人 (TOYAMA, Yoshihito)
東北大学・電気通信研究所・教授
研究者番号：００２５１９６８

桜井 貴文 (SAKURAI, Takafumi)
千葉大学・理学系研究科・教授
研究者番号：６０１８３３７３

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

(3) 連携研究者

小林 直樹 (KOBAYASHI, Naoki)
東京大学・情報理工学系研究科・教授
研究者番号：００２６２１５５