

平成 30 年 6 月 28 日現在

機関番号：12608

研究種目：基盤研究(C) (一般)

研究期間：2014～2017

課題番号：26330079

研究課題名(和文)汎言語的健全構文マクロ機構を用いた言語処理系構築手法の研究

研究課題名(英文)A methodology for programming language design through general hygienic macro expansion

研究代表者

脇田 建(Wakita, Ken)

東京工業大学・情報理工学院・准教授

研究者番号：10242265

交付決定額(研究期間全体)：(直接経費) 3,500,000円

研究成果の概要(和文)：本研究はプログラミング言語の複雑な構文処理実装をマクロと呼ばれる簡易な記述から自動生成する技術についての研究です。従来のマクロ技術に比べて、マクロ機構の働きが理解しやすい健全性を担保するとともに、核となる既存のプログラミング言語の構文もマクロで記述する2点において、この研究の先進性があります。応用面では、関数型リアクティブパラダイムを利用した組込み機器、社会シミュレーション分野に特化したプログラミング、グラフィックスや情報可視化における数式処理とドキュメントの自動生成などを研究しました。

研究成果の概要(英文)：The research aimed at development of a framework for designing and implementing a new programming language frontends by means of a macro-based technology. The novelty of the proposed macro technology are twofold: incorporation of hygienic macro system and description of the syntax of the core programming language. Based on this technology, we have conducted research in several application areas including functional reactive programming targeted for programming embedded systems, domain specific languages for social simulation, automated code and document generation system for graphics and information visualization domains.

研究分野：情報学

キーワード：ソフトウェア プログラミング メタプログラミング マクロシステム Hygienic構文マクロ ドメイン特化型言語 シミュレーション言語

1. 研究開始当初の背景

健全な構文マクロ機構は、オブジェクト指向言語のクラス構文のような複雑な構文記述を可能とし、素朴なマクロ機構に潜む落とし穴を回避する自動的な仕組みを内包している。本研究は、健全な構文マクロシステムが構文を拡張する仕組みをマクロ機能だけでなく、核言語の構文記述に応用することで、洗練されたマクロ機構を言語非依存に実装できることを示す。さらに、応用研究として、プログラミング言語だけでなく定理証明系、エージェント仕様記述にも挑戦し、マクロ技術の適用範囲を大きく広げることがめざす。

2. 研究の目的

本研究は、プログラマが目的に応じてプログラミング言語の構文を自由に拡張できる仕組みを対象とする。Cのマクロに代表される字句マクロシステムに対して、Lisp文化で培われた構文マクロ機構は構文木の変換をする高度なシステムである。例えば、Common Lispのオブジェクトシステム(CLOS)の構文は構文マクロで記述されている。この機構の研究は意図しない変数束縛を生じない(hygienic)構文マクロ機構とその効率的実装方式として90年代に一応の完成を見た。

利便性と実用性は認められたものの、健全な構文マクロ機構を取り込んだプログラミング言語は構文構造が簡素なLISPに限られており、一般のプログラミング言語での利用が望まれている。われわれは、LISPとの相互変換を中心とした提案により、一般のプログラミング言語においても比較的容易に健全な構文マクロ機構を構築できることを示した。本提案では、これまでの知見を踏まえ、以下の特徴をもつプログラミング言語独立な構文マクロ機構についての研究をする。

汎言語性 文法、意味論に依存せずプログラミング言語一般に適用可能

無からの拡張 構文解析器の自動生成機能を用いた核言語の構文解析器の自動生成

軽量変換器 Schemeとの相互変換にもとづく軽量の汎用マクロ変換器

定理証明系 定理証明系、仕様記述言語などの汎用言語以外の応用への挑戦

汎言語性構文マクロ機構を用いてプログラマは「拡張構文 核言語の構文」というプログラムの変換をパターンマッチを用いて記述する。たとえば右図は、JavaScriptに旅程を簡易記述するための**旅程**マクロの例だが、ここで導入される**旅程**の構文は核の構文には含まれないため、核言語の構文解析器はこの構文を解析することはできない。したがって、構文マクロ機構を利用するためには、その言語の構文解析器に拡張性を与えるこ

とが本質的に重要である。本研究は、この拡張性を核言語のプログラミング言語の文法、意味論から独立な形式で与える系統的な手法を明らかにしたい。過去の研究においては、JavaScriptを対象とし、マクロ定義を解析表現文法に変換し、そこから構文解析器を自動生成する手法を確立した。マクロ構文の構文解析の問題を言語独立な枠組みとして解決することが本研究の主目的の一つである。

無からの拡張 すでに述べたように、構文マクロ機構はマクロ定義から構文解析器を生成する能力を有している。ただし、このためには核言語の構文を解析表現文法で記述することが求められる。では、この面倒な作業を核言語の構文のマクロ記述で置き換えられないだろうか？前述の例の最後がwhile文に相当するマクロ定義であり、「while文の構文 組込み構文」という記述になっている。この記述から、マクロ機構はwhile文の構文解析器を生成できるはずである。われわれのJavaScriptのマクロ機構の実装において、半分以上核言語の文法記述が占めている。ここに提案した手法を利用することで構文マクロ機構の実装の大幅な簡素化が期待できる。この手法が成功すれば、構文をもたない原始言語に段階的に核言語の構文をマクロ定義するマクロ機構の斬新な実装手法となる。

相互変換 構文とその拡張の問題と同程度の重みをもつ課題がマクロ展開器の実装である。われわれの過去の研究では、右図のようなScheme処理系を介したマクロ展開方式を提案しており、そこでは核言語の構文木とS式の相互変換技術に重点をおいていた。これまでに扱った核言語はいずれもクロージャを有し、局所変数束縛を許すものであった。本研究では、この条件に該当しない言語についても(核言語ではなく)展開後のS式に対してクロージャ変換を施す方法によって、核言語への依存性を排除したマクロ展開方式を明らかにしたい。

仕様記述系 構文拡張が求められる場面はプログラミングだけではない。たとえば、集合論における内包記法、計算機科学における証明木、物理学の微分演算子など分野に特化した記法は珍しくない。ソフトウェアの仕様設計、検証、証明の場面では、当該ソフトウェアのドメインに特化した記法が必要とされる。本研究のマクロ機構の応用面の評価を行うために、特に定理証明系Coq、エージェント仕様記述系SOARSを対象としてマクロ機構を用いたユーザ定義構文の記述に挑戦する予定である。Coqにおいては演算子順位文法の拡張による構文定義機構が利用されている。この機構はコンパイラ実装技術にもとづくものであり、一般のユーザにとって心地良いとは言い難い。また、SOARSはマルチエージェントシミュレーションシステムである。

社会学者、政策立案者にも容易にシステムの記述を可能にすることは重要である。

3. 研究の方法

研究の基盤となる言語非独立なマクロ機構の開発は代表者が中心となって実施する予定である。メタプログラミングシステムとソフトウェア仕様の専門家である渡部、およびマルチエージェント社会システムの専門家である佐々木がそれぞれ定理証明系 Coq とシミュレーション記述言語 SOARS についてマクロ言語の分析・評価を担当し、脇田がその結果を受けて各言語のマクロ機能を提供する形で共同研究を進める予定である。平成 26 年より年次を経るにつれ、関数型言語、より一般のプログラミング言語、定理証明系・仕様記述言語へとマクロシステムの応用範囲を拡大する。

本研究の概要、年次計画、研究協力体制は以下のとおりである。研究の基盤は平成 26 年度に研究分担者と共同で実施する原始言語からの構文マクロ技術を用いた言語処理系の段階的構築法に関する研究である。この年度には、次年度から開発を本格化させる定理証明系、および仕様記述言語への構文マクロ技術の応用を見据えた調査を実施する。

平成 26 年度 原始言語からの言語処理系の段階的構築法など

われわれが既に開発しているマクロシステムは JavaScript に特化したものである。平成 26 年度は、核言語の記述もマクロシステムに取り込むことにより、構文マクロシステムの実装から言語に特化した箇所をなくすことにあたる。この年度の研究成果としては、言語非依存の構文マクロシステム、つまり、望みのプログラミング言語で構文マクロシステムを利用できることとなる。

この研究の核となる構文マクロシステムのうち、マクロ定義から解析表現文法を生成し、Scheme との相互変換により既存のマクロ展開器を再利用する方式については開発済みである。したがって、対象核言語の標準構文を構文マクロシステムで記述することが研究の中心となる。

構文マクロシステムの開発と並行して、マクロシステムの記述性の評価を進める。この目的のために、いくつかのプログラミング言語 (Java, Python, Scala)、定理証明系 (Coq)、エージェント仕様記述言語 (SOARS) などを対象とした記述実験を行う。

平成 27 年度 各種言語への構文マクロシステムの展開

本研究が対象とする、健全性 (Hygienity) は、関数閉包 (クロージャ) を用いて実装されるため、C, C++, Java のように関数閉包の機能に強い制約が課せられた言語でマクロシステムの健全性を担保する手段は自明ではない。この問題について、Scheme との相

互変換を利用している本提案においては、マクロを展開したあとの Scheme の式にクロージャ変換を施すことで、言語非依存な形で解決することを目指す。

前年度に開発をすませた構文マクロシステムのプロトタイプを用いて、構文マクロシステムとの親和性が高い関数型プログラミング言語について、構文マクロシステムの開発を進める。

およびエージェント仕様記述言語 SOARS 向けのマクロシステムの設計と実装を進める。

平成 28 年度 定理証明系・仕様記述言語向けのマクロ機能の提供

構文マクロシステム、および言語非依存なクロージャ変換器を利用して、クロージャの概念を提供しないプログラミング言語への構文マクロ機構の実装をする。言語としては Java を想定している。

これまでに開発した、プログラミング言語、定理証明系、およびエージェント仕様記述言語のマクロシステムについて、ユーザテスト、および評価を実施する。

4. 研究成果

平成 26 年度は基盤技術の開発を進めるとともに、渡部、佐々木は応用研究を実施し、特に並行オブジェクト計算における文脈の取り扱い、Functional Reactive Programming パラダイム、Web アプリケーションの UI テスト、エージェントシミュレーション向けブロック型ビジュアル言語等の応用分野についての研究を通してドメインに特化した記述についての考察が深化している。

平成 27 年度は基盤技術の開発においては健全構文マクロ機構の開発、およびその技術展開としてのドメイン特化型言語への拡張記述システムなどの応用研究を実施した。ここにおいては、マクロ機能を利用して構文拡張する方式で言語非独立なマクロシステムの開発を進めた。

応用方面の研究においては、並行オブジェクト計算における文脈の取り扱い、Functional Reactive Programming パラダイム、Web アプリケーションの UI テスト、エージェントシミュレーション向けブロック型ビジュアル言語等の応用分野についての研究を通してドメインに特化した記述についての考察を深化させた。

平成 28 年は基盤的な研究については、(1) マクロ定義のパターンとテンプレートの対比を字句レベル構文解析 (Parsing Expression Grammar) を用いたコンパイラ生成器で実装し、構文マクロ展開後の抽象構文木をアンパースしてマクロ記述を含まないホスト言語のコードを生成するアプローチ (伊藤、脇田) と (2) 抽象字句ストリーム上での構文解析文法処理システムの OMeta に近

年 Flatt が提案したスコープ集合を応用したマクロ展開方式を応用したアプローチ(星野、高桑、渡部)を試みた。

応用面では、以下のいくつかの研究を実施した: (3) GPGPU を用いた汎用並列計算や実時間組込みプログラミングについて応用領域特化型プログラミング技術を応用した Python の拡張言語を開発(鎌田、佐々木) (4) プログラミングが困難な実時間組込みシステムについて関数型リアクティブプログラミングを実施できるメタプログラミング機構を設計し可読性とデバッグの容易性を高めた。(中野、佐々木)、(5)メタプログラミングを用いた情報可視化システムの自動生成の研究(高野、武田、脇田)、(6)メタプログラミングを用いたソフトウェアドキュメントの自動生成に向けての一般的な手法(高野、武田、脇田)を研究した。

平成 29 年度は字句レベル構文解析文法、および抽象字句ストリームを対象とした構文解析文法を用いる手法について研究をさらに実施した。(1) 字句レベルの手法についてはさまざまなプログラミング言語の変数束縛の意味論の違いを表現する手法についての研究を薦めた。(2) 抽象字句ストリームの手法については前年度に研究した OMeta でスコープ集合を扱うマクロ展開方式(星野、高桑、渡部)の完成度を高めた。

応用面では、以下のいくつかの研究を実施した。(3) 高野、脇田はグラフィックプロセッサ向けの高速度描画機能の実装に数式処理システムを用いたメタプログラミングを応用する手法を発表し、インタラクティブな可視化システムの高速度化に成功した。(4) 中野と佐々木は前年度の研究を発展させ関数型リアクティブプログラミングのメタプログラミング機構を元にドメイン特化型言語を提案した。(5)山梨、澤入、佐々木は初学者向けのブロック言語にメタプログラミングを応用し、システムとして実装した。

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 8 件)

1. 高桑健太郎、渡部卓雄、多言語に対応した衛生的マクロ機構導入方式、コンピュータソフトウェア 35(2), pp. 33-39, 2017. (査読あり)
2. 脇田建、複雑系ネットワークの可視化～社会ネットワークを中心に～、オペレーションズ・リサーチ (63), pp. 13-19, 2018. (査読なし、招待論文)

[学会発表](計 32 件)

1. Kensuke Sawada and Takuo Watanabe,

Enfrp: A functional reactive programming language for small-scale embedded systems, Modularity 2016. (査読あり)

2. Shohei Yasutaka and Takuo Watanabe, Actario: A framework for reasoning about Actor system, workshop on Programming based on Actors, agents, and decentralized control, 2015. (査読あり)
3. 脇田建, (数式 実装、文書): OpenGL 計算ライブラリ実装の経験から, SIGPX 2015. (査読なし)
4. 山梨裕矢、佐々木晃、エージェントシミュレーション向けブロック型ビジュアル言語の試作、計測自動制御学会, 2016. (査読なし)
5. 星野友宏、高桑健太郎、渡部卓雄、OMeta のための衛生的マクロ定義機構導入方式、JSSST 2016. (査読なし)
6. 脇田建、高野陸、武田一起、数学的記号処理システムを用いたソフトウェアの構成手法、JSSST 2016. (査読なし)
7. 鎌田知也、佐々木晃、Python を拡張した GPGPU 向け言語開発環境の設計、第 15 回情報科学技術フォーラム、2016. (査読なし)
8. 伊藤玲於奈、脇田建、汎言語的 Hygienic 構文マクロシステムの実装に向けて、第 58 回プログラミング・シンポジウム、2017. (査読なし)
9. 高野陸、脇田建、数式処理システムを用いた対話的高次元グラフ可視化方式の実装、JSSST 2017. (査読なし)
10. 高桑健太郎、渡部卓雄、多言語に対応した衛生的マクロ機構導入方式、第 19 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、2017. (査読あり)
11. 中野史彬、佐々木晃、関数型リアクティブプログラミングによる組込みシステム向け DSL の実装、IPSJ 2017. (査読なし)
12. Kazuhiro Shibana and Takuo Watanabe, Actoverse: A reversible debugger for Actors, workshop on Programming based on Actors, agents, and decentralized control, 2017. (査読あり)
13. Takuo Watanabe, Reactive reflection

in an FRP language for small-scale embedded systems, workshop on Meta-programming techniques and reflection, 2017. (査読あり)

14. 山梨裕矢、佐々木晃、ビジュアルプログラミングとテキスト記述形式の連携による初学者向けプログラミング学習システムの提案、第 59 回プログラミング・シンポジウム、2017. (査読なし)
15. 澤入圭佑、佐々木晃、ビジュアルブロックの自動生成を特徴としたブロック言語処理系開発システムの提案、第 59 回プログラミング・シンポジウム、2017. (査読なし)
16. 高野陸、脇田建、対話的なネットワーク可視化分析のシェーダ実装による高速化、JSSST 2017. (査読なし)
17. Ken Wakita and Riku Takano, Interactive high-dimensional visualization, Handai IDS-JFLI joint workshop on Media and Graphics, 2018. (査読なし、招待講演)
18. Riku Takano and Ken Wakita, Fluid UI for High-dimensional analysis of social networks, IUI 2018. (査読あり)

〔図書〕(計 件)

〔産業財産権〕

出願状況(計 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 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
脇田建 (Wakita, Ken)
東京工業大学・情報理工学院・准教授
研究者番号：10242265

(2)研究分担者
渡部卓雄 (Watanabe, Takuo)
佐々木晃 (Sasaki, Akira)