

## 科学研究費助成事業 研究成果報告書

令和 2 年 7 月 7 日現在

機関番号：62615  
研究種目：若手研究(B)  
研究期間：2016～2019  
課題番号：16K16040  
研究課題名(和文)リアルタイム型エラーデバッグの基礎理論と実際

研究課題名(英文) Realtime type error debugging

## 研究代表者

対馬 かなえ (Tsushima, Kanae)

国立情報学研究所・アーキテクチャ科学研究系・特任助教

研究者番号：80754663

交付決定額(研究期間全体)：(直接経費) 2,900,000円

研究成果の概要(和文)：リアルタイムに型エラーを修正(デバッグ)するための手法について研究を行った。まず一つ目はプログラム記述中の構文解析である。これには実装のコストを削減するために、既存の構文解析器を使用する手法を提案した。二つ目は型エラーの修正手法の改良である。主な成果は(1)エラーのある範囲を絞り込む型エラースライサの実用的な改良手法(2)エラー範囲が絞り込まれた型エラースライスを複数個使用してより良い情報を見つける手法(3)いくつかの型エラーデバッグ手法を統合的に扱うフレームワーク(4)既存の型推論器を使用してエラーメッセージのように二つの矛盾する型を示す手法、などである。

## 研究成果の学術的意義や社会的意義

プログラムは現代社会を支える基盤の一つとなっており、プログラムの安全性の確保や、作成するためのコストの削減は重要である。本研究では、型によってある種の安全性が保証されるプログラムを、プログラマがこれまでより容易に記述できる方法について研究した。それにより、安全性の高いプログラムが増えると同時に、プログラマへの負担を軽減することが期待される。

研究成果の概要(英文)：In this work, we have investigated real-time type error debugging. First, we have developed parsing of programs in progress. For this, we use an existing parser for reducing the cost of implementation. Second, we have improved existing type error debugging approaches. The followings are results: (1) Improved type error slicing approach (narrowing the area of possibility of type errors) (2) Developed a novel approach for finding useful information using multiple type error slices (3) Developed a novel approach of a framework that enables several type error debugging approaches (4) Developed an approach that finds counter-factual type error messages using type inferencer.

研究分野：プログラミング言語

キーワード：型エラー デバッグ 関数型言語 型エラースライス

## 1. 研究開始当初の背景

プログラミング言語において、型はプログラムの安全性を保証する道具として広く使用されている。型の整合性を検査することで、数字と文字列の引き算 (1 - "apple") のように実行結果が決まっていないプログラムを排除して、正しく安全に実行が出来るプログラムだけを残すことができる。しかし、正しく型がつくプログラムを書くことは容易ではない。プログラムに型がつかない場合、プログラマはコンパイラのエラーメッセージを参考に型エラーの原因を探ることになるが、型エラーメッセージが役に立たないことも多い。プログラマが型エラーデバッグに多くの時間を費やすことを避けるために、型エラーのデバッグ手段の提供は重要である。加えて、エラーが複数存在すると修正がさらに難しくなる・プログラム中の間違いが他の間違いを誘発するといった問題が存在する。それらを回避するには、エラーが出現したならばすぐに修正する手段、すなわちプログラムの記述と並行した型エラーのデバッグ (リアルタイム型エラーデバッグ) が必要である。しかし、現在、残念ながらほとんどの言語で提供されていない。

これは型の付いた信頼性の高いプログラムを書くことを妨げ、さらには型の付いた言語の使用を諦め、その結果、プログラムの信頼性を落とす結果にもなりかねない。

## 2. 研究の目的

ユーザが状況と必要に応じてリアルタイムな型エラーデバッグ支援を受けることのできるフレームワークを構築するために、次の2つの目標に分けてこれらを実現する。[リアルタイムな構造解析] では、既存手法で困難なキーワードの少ない言語でのリアルタイム構造化手法を確立し、実装を行う。得られた構造化プログラムに型エラーがある場合には [各種型エラーデバッグ] を行い、記述時のデバッグを可能にする。

## 3. 研究の方法

[リアルタイムな構造解析] 実際に広く使用されている言語を対象として行うのは実用的な言語では構文等が多く、困難である。そこで、“コンパイラの機能を利用”・“プログラムが変更される前の構造を利用” というアイデアを用いて打開する。具体的にはプログラムが変化した場合、コンパイラの字句解析・構文解析を部分的なプログラムに用い、その部分だけを字句・構文解析にかけることで構造化が可能か否かを判断する。構造化が可能なら変化する前のプログラムの構造に埋め込んだ形で用い、不可能なら未構造化部分として残すことで、それ以外の部分の構造を維持する。コンパイラの字句・構文解析を部分プログラムに対して使用することで、実装の労力を抑えつつ、柔軟なリアルタイム構造化が実現できる。

[各種型エラーデバッグ] 構文解析で得られるリアルタイムなプログラム構造と記述終了後の構造の違いは、前者に空白部分・未構造化部分が存在することである。そのままでは型エラーデバッグ手法を適用することはできない。そこで、それらを型制約を持たない無害なコードに置き換えることで、完全なプログラムにするというアイデアを用いる。これによってこれまでの型エラーデバッグ手法への適用が可能になる。加えて、これまでの型エラーデバッグ手法の改良や、新しい型エラーデバッグ手法の提案を行う。

## 4. 研究成果

[リアルタイムな構造解析] 漸進的構文解析のように、一部だけ変更されたプログラムに対して構文解析を行う手法を提案した。漸進的構文解析との違いとしては、既存のコンパイラの構文解析器を使用して行う点である。そのため、特殊な構文解析器を作成する必要がなく、既存の言語を対象に一部だけ変更されたプログラムに対する構文解析の実装が容易になった。具体的には、コンパイラの構文解析器を使用して、構文同士の結合度による組み替え・他の構文の略奪の規則を導出し、それらが行われるプログラムを自動的に生成する。括弧等の抽象構文木の時点で失われる情報は人が補う必要があるが、それ以外のプログラムは扱うことができた。

### [型エラーデバッグ]

型エラーデバッグ手法の改良・新しい型エラーデバッグ手法の提案では四つの研究を行った。

(1) 型エラースライシングの高速化：これまで扱ってきた型エラースライシング手法では、問題点が二つあった。一つ目は抽象構文木の上の方から一箇所ずつ削って行くという手法を採用していた点である。しかしそれではリストのような深い場所が大きく偏った構造に対して大きさに比例した時間がかかってしまう。これに対しては、おおよそ半分ずつ削ることができるような改良を加えることで、高速化を実現した。二つ目は、スライシングにあたってプログラムの一部を例外の発生の構文に変換してコンパイラの型推論器にかけ型エラーかどうかの判定を行っているが、プログラムの多くの部分をそのように置き換えると、型推論に時間のかかるプ

ログラムになってしまうことが多い点である。この点を改善するために、これまでの単純に置き換えたプログラムから、より型推論に時間のかからない同等なプログラムに変換する手法を提案した。

(2) 複数の型エラースライスを用いたデバッグ：既存の型エラースライサを使用して、複数の型エラースライスを求める手法を提案した。それにより、より型エラーの原因である可能性が高い箇所を特定できるようになった。

(3) 型エラーデバッグ手法を統一的に扱うフレームワークの導入：型エラーのデバッグでは、実際の型エラーのプログラムやそれをデバッグするユーザによって、有効な手法が異なる。これまでの手法では、型エラースライサ・型エラーデバッガ・型エラーメッセージの生成等にはそれぞれ独立した実装が必要であった。本研究では、プログラム中のある一部のみの principal type と、その部分に対して周りから期待される型からなるような木である、Type debugging information tree (TDIT) を導入することで、それらを統一して扱うことができるようになった。

(4) プログラム変換による複数の型エラーメッセージの生成：コンパイラの型エラーのメッセージは型推論の順序に依存しており、一つのプログラムに対して一つしか得られない。本研究では型エラーのプログラムに対してプログラム変換をかけることで推論の順序を変え、複数の型エラーメッセージを得る手法を提案した。またこの手法では、可能な限り変換しつつ型エラーメッセージを集めていくことで、同時に型エラースライスを求めることができる。

## 5. 主な発表論文等

〔雑誌論文〕 計4件（うち査読付論文 3件 / うち国際共著 0件 / うちオープンアクセス 0件）

1. 著者名 対馬 かなえ、佐藤 重幸	4. 巻
2. 論文標題 複数の型エラースライスによる ill-typed プログラムの分析	5. 発行年 2018年
3. 雑誌名 第20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	6. 最初と最後の頁 17ページ
掲載論文のDOI (デジタルオブジェクト識別子) なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 脇川 奈穂、対馬 かなえ	4. 巻
2. 論文標題 実用的な型エラーライサの提案と評価	5. 発行年 2018年
3. 雑誌名 第20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	6. 最初と最後の頁 17ページ
掲載論文のDOI (デジタルオブジェクト識別子) なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 Kanae Tsushima, Olaf Chitil	4. 巻
2. 論文標題 A Common Framework Using Expected Types for Several Type Debugging Approaches	5. 発行年 2018年
3. 雑誌名 Fourteenth International Symposium on Functional and Logic Programming	6. 最初と最後の頁 12pages
掲載論文のDOI (デジタルオブジェクト識別子) なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 Kanae Tsushima, Olaf Chitil, Joanna Sharrad	4. 巻
2. 論文標題 Type Debugging with Counter-Factual Type Error Messages Using an Existing Type Checker	5. 発行年 2020年
3. 雑誌名 Proceedings of the 31st Symposium on Implementation and Application of Functional Languages	6. 最初と最後の頁 12pages
掲載論文のDOI (デジタルオブジェクト識別子) なし	査読の有無 無
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

〔学会発表〕 計7件（うち招待講演 0件 / うち国際学会 3件）

1. 発表者名 対馬 かなえ、佐藤 重幸
2. 発表標題 複数の型エラースライスによる ill-typed プログラムの分析
3. 学会等名 第20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
4. 発表年 2018年

1. 発表者名 脇川 奈穂、対馬 かなえ
2. 発表標題 実用的な型エラースライサーの提案と評価
3. 学会等名 第20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
4. 発表年 2018年

1. 発表者名 対馬 かなえ、脇川 奈穂
2. 発表標題 実用的な型エラースライサーに向けた改良と評価
3. 学会等名 日本ソフトウェア科学会第 34 回大会
4. 発表年 2017年

1. 発表者名 Kanae Tsushima, Olaf Chitil
2. 発表標題 A Common Framework Using Expected Types for Several Type Debugging Approaches
3. 学会等名 Fourteenth International Symposium on Functional and Logic Programming (国際学会)
4. 発表年 2018年

1. 発表者名 Kanae Tsushima
2. 発表標題 A semi-embedded incremental parsing
3. 学会等名 The 28th symposium on Implementation and Application of Functional Languages (IFL 2016) (国際学会)
4. 発表年 2016年

1. 発表者名 対馬かなえ
2. 発表標題 既存の構文解析器を利用した漸進的構文解析
3. 学会等名 情報処理学会 58回プログラミング・シンポジウム
4. 発表年 2017年

1. 発表者名 Kanae Tsushima, Olaf Chitil, Joanna Sharrad
2. 発表標題 Type Debugging with Counter-Factual Type Error Messages Using an Existing Type Checker
3. 学会等名 The 31st Symposium on Implementation and Application of Functional Languages (国際学会)
4. 発表年 2019年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

-

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考