

科学研究費助成事業 研究成果報告書

平成 30 年 5 月 25 日現在

機関番号：11301

研究種目：基盤研究(B) (一般)

研究期間：2013～2017

課題番号：25280019

研究課題名(和文) ML系多相型言語SML#の実用化技術に関する基礎研究

研究課題名(英文) Basic research on implementation technology for making SML# a practical polymorphic language

研究代表者

大堀 淳(OHORI, Atsushi)

東北大学・電気通信研究所・教授

研究者番号：60252532

交付決定額(研究期間全体)：(直接経費) 6,300,000円

研究成果の概要(和文)：ML系多相型言語SML#の実用化技術に関する基礎研究を行い、(1)汎用の低レベル言語用コード生成方式の構築およびLLVM向けのバックエンドの開発、(2)並行GC方式とアルゴリズムの構築およびマルチコアにスケールする並行GCシステムの開発と実装、(3)部分動的レコードを用いたJSONデータ操作方式の開発と実装、(4)産業界と連携した実用ERPシステムの開発、等の成果をあげ、これら成果をSML#コンパイラに実装し、世界に向けて継続的にリリースした。また、これら成果を含むSML#コンパイラの体系的なテスト環境を構築し、コンパイラの堅牢性を高めた。

研究成果の概要(英文)：We conducted basic research on implementation technology for making SML# a practical polymorphic language. Major achievements include the following: (1) the development of code generation method for general purpose low-level code language and an LLVM back-end for SML#, (2) fully concurrent GC algorithm for multicore CPUs and its implementation, (3) SML# JSON support based on the type system of partially dynamic records, (4) a practical ERP system development with industry. We have implemented those achievements in the SML# compiler and have released the compiler. To enhance the robustness of the SML# compiler, we have also developed a systematic test environment of the SML# compiler.

研究分野：ソフトウェア

キーワード：SML# コンパイラ 実装方式 最適化 コード分析

1. 研究開始当初の背景

プログラミング言語 SML#は、報告者等による ML 系多相型関数型言語の基礎理論研究成果を結集し設計された言語であり、その研究開発を通じて確立されたレコード多相性の理論[引用文献 1, 2]やデータベース問い合わせ言語 SQL との統合[引用文献 3, 4]などを含む種々の理論や技術の幾つかは多相型言語の設計やコンパイル方式等の研究の先駆的な成果である。しかしながら、このような先端機能を装備した SML#も、実際に産業界で実用システムを開発するのに十分な完成度を持っているとは言えない。関数型言語の実装技術研究の現状では、これら実用機能を系統的に実現する方法論が確立されているとはいえない。さらに、型主導コンパイルを実現し従来不可能であった高度な機能を実現している SML#などのような複雑なコンパイラに対するアドホックな改良は困難である。この状況は、高度な型主導コンパイルにより先端機能を実現する多相型言語の実用化技術の確立には、その言語が基礎とする型理論にまで遡る種々の新たな理論や技術が必要であることを示唆している。

2. 研究の目的

SML#は、レコード多相性、C との直接連携、データベース問い合わせ言語 SQL のシームレスな統合、分割コンパイルとリンク等、実用システム開発にとって重要な機能を取り入れた次世代関数型言語である。これら SML#の先端機能は、申請者等によるデータベースの型理論や型主導コンパイル方式を通じて達成されている。この先端性のため、関数型言語に対する従来の分析手法等の適用が困難であり、現状では、最適化技術や開発ツール構築技術などが十分に確立していない。それら実用化技術の確立には新たな基礎理論研究が必要である。本研究では、この潜在的に高い実用性をもつ ML 系関数型言語である SML#を、我が国、ひいては世界の高度なソフトウェア生産現場にて十分に使える言語として完成させるために、最先端型主導コンパイラの実用化技術確立を目指した基礎研究を実施する。

3. 研究の方法

前述の通り、SML#のような高度な型理論的な枠組みを基礎として設計されたプログラミング言語の高速化には、従来アドホックなコンパイラの最適化技術が適用できないことが多い。たとえば、我々が新たに構築し SML#に実装された型主導コンパイル方式は、レコード多相性などの従来不可能であった種々の実用機能の実現を可能にしたが、その反面効率の悪いコードとなる恐れがある。また、高階の関数型言語である SML#にとって必須の自動メモリー割当機構(ガーベージコレクション、GC)は、マルチコア上での軽量の並列スレッドの実行と相性が悪く、そのため、

従来の ML 系関数型言語では、マルチコア上の軽量の並列スレッドの実行が十分にサポートされていない。さらに、高階の型付き言語は、動的な言語に比較して外部データの扱いが困難である。本研究では、これら問題に対応するために、従来のコンパイル方式を型理論的な枠組みを用いて分析し、コンパイルおよび操作的意味論に関する新たな枠組みを構築し、その理論的な枠組みを基礎に効率よい実装方式の構築を行い、さらに、それらを SML#コンパイラに実装し、その有効性を確認する戦略を取る。

4. 研究成果

上記の戦略の下、新たな理論の構築からコンパイラの実装に亘る研究を実施し、多数の新たな知見やシステムの開発に成功した。以下その主なものを報告する。

(1) 汎用の低レベル言語用コード生成方式の構築および LLVM バックエンドの開発:

型主導コンパイル方式を実現する SML#は、従来のコード生成方式とは相容れない点が多い。本研究では、LLVM バックエンドの意味論分析とそれに基づく SML#への接続可能性を分析し、さらにそれらを基礎に LLVM バックエンドと整合性ある関数呼び出しに関する型主導コンパイル方式を構築し、それらを取り入れた SML#コンパイラの LLVM バックエンドの開発を成功させた。さらに、そのバックエンドを 64 ビットアーキテクチャへ拡張し、幅広いプラットフォームで動作可能な基盤の構築に成功した。この成果は、2014年の ACM ML ワークショップにて発表するとともに、SML#に実装し、世界にリリースした。

(2) マルチコア対応の並列 GC 方式の構築と実装:

SML#を含む関数型言語にて並列スレッドをサポートする上での最大の困難は、並列スレッドと並行に動作する効率よい GC アルゴリズムの構築と実装である。この問題は、世界的にみても、十分に満足いく解決策が提案されていない重要な課題である。本研究では、この問題に取り組み、我々がすでに開発しているオブジェクトを動かさない GC を Yuasa のスナップショット GC 技術と統合することによって、マルチコア上で動作し、ユーザスレッドと完全に並行に動作する GC アルゴリズムの開発に成功し、さらにその実装技術を構築し、SML#に実装し、その実用性を確認した。この成果は、2016年の ACM ICFP で発表された。また、この GC を装備した SML#コンパイラを世界にリリースした。

(3) 部分動的レコードを基礎とする JSON データアクセス機構:

ML 系言語は、その堅牢な静的型システムのために、型無しのデータである外部データの柔軟な扱いが困難である。この問題に対処する

ため、報告者等が以前に構築した部分動的レコードの型理論を拡張し、さらにその型理論を基礎に、現在 WEB アプリケーションなどで広く使用されている外部データ表現の一つである JSON データを安全かつ高水準に扱うことを可能にする言語機構を開発し、その実装技術を構築し、SML#コンパイラに実装した。この成果は、オブジェクト指向計算のトップコンファレンスの一つである ECOOP2016 で発表され、また、この機構を装備した SML#コンパイラを世界にリリースした。

(4) 産学連携 ERP システム開発:

関数型言語は高水準な記述能力と静的型システムによる信頼性から、ソフトウェアの生産を向上させると期待されながら、産業界にあまり普及していなかった。その原因の一つは、データベースや OS との直接連携の困難や外部データの扱いにくさにある。本研究の成果もふくめ、これら機能が大幅に強化された SML#言語は、産業界のソフトウェア生産に貢献すると期待できる。この洞察の下、NEC ソリューションズイノベータ殿の協力の下、SML# を用いた実用的なソフトウェア開発プロジェクトを、ERP システムを例として、実施し、学術研究の成果である SML# の実用性の検証を行った。その結果、SML# の先端機能の有用性が確認された。この開発プロジェクトの詳細は、関数プログラミングのトップコンファレンスである ACM ICFP2014 で報告された。学術研究の成果である大学発の新しいプログラミング言語を用いた産学連携のソフトウェア開発プロジェクトの報告が ACM ICFP に採録されるのは、我が国のプログラミング言語の基礎研究にとって前例のない快挙と言える。

(5) SML#コンパイラのリリース:

上記の研究成果の各項目でも言及した通り、上記を含む研究成果は、ML 系言語のフルスケールコンパイラである SML# に実装され、オープンソースとして世界に提供している。本研究期間に限っても 2014 年 4 月の 2.0.0 版から 2017 年 4 月の 3.4.0 版にいたるまで、世界にむけた 7 回のリリースを行った。さらに、本研究では、それら新しい機能を含む SML# 言語コンパイラの体系的なテスト環境を構築し、SML# の堅牢性を高めた。

< 引用文献 >

1. P. Buneman, A. Ohori. Polymorphism and Type Inference in Database Programming. ACM Trans. Database Syst. 21(1): 30-76 (1996)
2. A. Ohori. A Polymorphic Record Calculus and Its Compilation. ACM Trans. Program. Lang. Syst. 17(6): 844-895 (1995)
3. A. Ohori. A Compilation Method for ML-Style Polymorphic Record Calculi. ACM POPL 1992: 154-165.
4. Ohori, P. Buneman, V. Tannen. Database Programming in Machiavelli - a Polymorphic Language with Static Type Inference. ACM SIGMOD Conference 1989: 46-57
5. 主な発表論文等
〔雑誌論文〕(計 4 件)
 1. 逢坂美冬, 上野雄大, 大堀淳. 部分動的レコードを活用した型付きテンプレートエンジンの実現. コンピュータソフトウェア. 印刷中(2018). (査読あり)
 2. Katsuhiko Ueno, Atsushi Ohori. A Type Safe Access to Key-value Stores from Functional Languages. Journal of Information Processing Vol. 24, No. 1, pp.141-151, 2016. (査読あり)
 3. Katsuhiko Ueno, Atsushi Ohori. A foreign language interface from ML to shell. New Generation Computing, Vol. 34, No. 3, pp. 239--256, 2016. (査読あり)
 4. 遠藤誠典, 百足勇人, 森畑明昌, 上野雄大, 大堀淳. 変数参照関係を用いた関数型プログラムのコードリーディング支援. コンピュータソフトウェア 32, 194-212, 2015. (査読あり)
- 〔学会発表〕(計 23 件)
 1. 美馬久行, 上野雄大, 大堀淳. 多相関数を含むプログラムの抽象解釈を用いた最適化. 日本ソフトウェア科学会第 34 回大会. 慶応義塾大学日吉キャンパス. 2017 年 9 月.
 2. 大野一樹, 上野雄大, 大堀淳. SML# のためのコードレベルデバッグ環境の構築に向けて. 日本ソフトウェア科学会第 34 回大会. 慶応義塾大学日吉キャンパス. 2017 年 9 月.
 3. Atsushi Ohori, Kenjiro Taura, Katsuhiko Ueno. Making SML# a general-purpose high-performance language. ACM SIGPLAN Workshop on ML. Oxford University. 2017 年 9 月.
 4. Tomohiro Sasaki, Katsuhiko Ueno, Atsushi Ohori. SML# with Natural Join. ACM SIGPLAN Workshop on ML. 奈良春日国際フォーラム. 2016 年 9 月.

5. Katsuhiro Ueno, Atsushi Ohori. A Fully Concurrent Garbage Collector for Functional Programs on Multicore Processors. Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming (ICFP 2016), pp. 421--433. 奈良春日国際フォーラム. 2016年9月.
6. 佐藤友昭, 上野雄大, 大堀淳. SML#のよりシームレスな外部関数インターフェースの実現に向けて. 日本ソフトウェア科学会 第33回大会. 東北大学. 2016年9月.
7. 佐々木智啓, 上野雄大, 大堀淳. 自然結合制約を含む型推論アルゴリズムの実装方式. 日本ソフトウェア科学会 第33回大会. 東北大学. 2016年9月.
8. 美馬久行, 上野雄大, 大堀淳. SML#による Vertex-centric プログラミングに向けて. 日本ソフトウェア科学会 第33回大会. 東北大学. 2016年9月.
9. 逢坂美冬, 上野雄大, 大堀淳. 部分動的レコードを活用した型付きテンプレートエンジンの試作. 日本ソフトウェア科学会 第33回大会. 東北大学. 2016年9月7日.
10. 徳永航平, 上野雄大, 大堀淳. OS を関数型言語のみで開発するための検討と試作. 日本ソフトウェア科学会 第33回大会. 東北大学. 2016年9月.
11. Atsushi Ohori, Katsuhiro Ueno, Tomohiro Sasaki, Daisuke Kikuchi. A Calculus with Partially Dynamic Records for Typeful Manipulation of JSON Objects. Proceedings of the 30th European Conference on Object-Oriented Programming (ECOOP 2016), pp. 18:1-18:25. Trevi Congress Center, Rome, Italy. 2016年7月.
12. 新田祐児, 上野雄大, 大堀淳. 関数型と組型がネストした型を持つ変数を含んだ式の自動生成手法. 情報処理学会プログラミング研究会 第108回プログラミング研究発表会. 東京大学. 2016年2月.
13. 田畑憲太, 上野雄大, 大堀淳. コンパイラ実装言語で中間表現データ構造を記述するための言語機構. 情報処理学会プログラミング研究会 第108回プログラミング研究発表会. 東京大学. 2016年2月.
14. Katsuhiro Ueno. The SML# compiler backend: compiling ML to C-compatible low-level code. Syntax and Semantics of Low-Level Languages (LOLA 2015), Invited talk. 京都大学. 2015年7月.
15. 上野雄大, 大堀淳. 関数型言語からキーバリューストアへの型安全なアクセス機構. 情報処理学会 第104回プログラミング研究発表会. 富山県教育文化会館. 2015年6月.
16. 逢坂美冬, 菊地大介, 上野雄大, 大堀淳, 佐々木加奈子. 関数型言語による高水準なWebアプリケーション開発環境. 情報処理学会 第104回プログラミング研究発表会. 富山県教育文化会館. 2015年6月.
17. Katsuhiro Ueno, Yutaka Fukasawa, Akimasa Morihata, Atsushi Ohori. The Essence of Ruby. Asian Symposium of Programming Languages and Systems, pp 78-98. National University of Singapore. 2014年11月.
18. Atsushi Ohori, Katsuhiro Ueno, Kazunori Hoshi, Shinji Nozaki, Takashi Sato, Tasuku Makabe, Yuki Ito. SML# in Industry : A Practical ERP System Development. ACM International Conference on Functional Programming, pp 167-173. Gothenburg, Sweden. 2014年09月.
19. Katsuhiro Ueno, Atsushi Ohori. Compiling SML# with LLVM a Challenge of Implementing ML on a Common Compiler Infrastructure. ACM SIGPLAN ML Family Workshop. Gothenburg, Sweden. 2014年09月.
20. 斎藤皓, 上野雄大, 森畑明昌, 大堀淳. SML#のSQL統合機能への行集約機能の実装. 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. 熊本県阿蘇市阿蘇温泉「阿蘇の司 ビラパークホテル」. 2014年03月.
21. 上野雄大, 大堀淳. 関数型言語からキーバリューストアへの型安全なアクセス機構. 日本ソフトウェア科学会第30回大会, 東京大学. 2013年09月.
22. 遠藤誠典, 百足勇人, 森畑明昌, 上野雄大, 大堀淳. 変数参照関係を用いた関数型プログラムのコードリーディング支援. 日本ソフトウェア科学会第30回大会. 東京大学. 2013年09月.

23. 大堀淳. 関数型言語 SML#の開発 : 我が国のソフトウェア産業新生へのささやかな貢献を目指して. 日本ソフトウェア科学会第 30 回大会 (招待講演). 東京大学. 2013 年 09 月.

〔その他〕

オープンソースソフトの世界への発信

リリース日 ソフトウェア

2014-04-04 SML# 2.0.0 版

2016-03-31 SML# 3.0.0 版

2016-04-04 SML# 3.0.1 版

2016-05-26 SML# 3.1.0 版

2016-07-15 SML# 3.1.1 版

2016-09-16 SML# 3.2.0 版

2017-06-20 SML# 3.3.0 版

2017-08-31 SML# 3.4.0 版

ホームページ等

SML# ホームページ :

<http://www.pllab.riec.tohoku.ac.jp/smls/harp/ja/>

6. 研究組織

(1) 研究代表者

大堀 淳 (OHORI, Atsushi)

東北大学・電気通信研究所・教授

研究者番号 : 60252532

(2) 研究分担者

上野 雄大 (UENO, Katsuhiko)

東北大学・電気通信研究所・准教授

研究者番号 : 60551554