

平成 26 年 6 月 7 日現在

機関番号：17104

研究種目：挑戦的萌芽研究

研究期間：2011～2013

課題番号：23650013

研究課題名(和文)汎用性に優れた型付中間言語

研究課題名(英文)Towards Highly Universal Typed Intermediate Languages

研究代表者

八杉 昌宏 (YASUGI, Masahiro)

九州工業大学・大学院情報工学研究院・教授

研究者番号：30273759

交付決定額(研究期間全体)：(直接経費) 2,700,000円、(間接経費) 810,000円

研究成果の概要(和文)：本研究は、仮想機械のRISC化を目標に汎用性と高速性を持つ型付中間言語の設計に挑んだ。低水準命令列に関する既存の依存型による型検査技法をベースとし、ビットレベル操作におけるfixnumの実現とシフト命令への対応を目標に場合依存型という新しい概念と二分決定グラフ(BDD)を、オブジェクトの初期化における循環データ構造への対応を目標にprovisional assumptionsという新しい概念を、高速な多次元配列操作への構造的対応を目標に添え字計算に関する新しい述語を、それぞれ検討できた。

研究成果の概要(英文)：In this study, we tried to design highly universal and fast typed intermediate languages by pursuing RISC-based abstract machines. On the basis of the existing techniques of type checking with dependent types for low-level instruction sequences, we examined (1) a new notion of case dependent types and binary decision diagrams (BDDs) for implementing fixnums and supporting the shift instructions in bit-level operations, (2) a new notion of provisional assumptions for supporting typed construction of cyclic data structures, and (3) a new predicate on subscript calculation for supporting fast multidimensional array operations in a structural manner.

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：型付中間言語 低水準言語 プログラミング言語処理系 型検査 多次元配列 循環データ構造 高性能計算

1. 研究開始当初の背景

近年の情報通信ネットワークの発展は、各地のソフトウェアの開発者と利用者とを結びつけ、コード(プログラム)の活発な配信も可能となっている。その際、バグや攻撃による被害をいかに防止して、安全性を確保するかという点が極めて重要であると考えられている。Java 仮想機械語(JVML)や Common Intermediate Language(CIL)に代表される既存の型付中間言語は、実行前型検査と自動メモリ管理(GC)によりそのプログラムを安全に実行できるが、型システムが特定のオブジェクト指向言語(Java や C#)にほぼ対応している点を含め汎用性は高くない。また、汎用性が高いはずの低水準言語であっても、Morrisett らの TAL (Typed Assembly Language), TALx86、に代表される既存研究では(一部を除き): Lisp 系言語の実装で典型的な fixnum、冗長な実行時チェックの除去済みコードへの型付け、正確な GC 等が十分にサポートされているとはいえなかった。

2. 研究の目的

本研究は、従来困難と考えられていた、汎用性の高い型付中間言語、すなわち広く様々な言語からの翻訳の共通のターゲットとしての利用に優れた低水準の型付中間言語 MIL を、安全で高性能な言語処理系(計算基盤)が作成できる形で設計することを目的とした。

具体的には仮想機械の RISC 化を目標に、RISC 命令の実行が許可できる制約を表す型環境という視点で型システムを構築することで、安全性を維持しつつビットレベル構造を扱う命令への型付けを含む圧倒的な自由度(汎用性)と高速性を持つ型付中間言語(以降、この言語を“MIL”と呼ぶ)を設計しようとするものである。低水準中間言語の既存研究が深く取り組んでいない正確な GC のためのトレーサビリティ保証にも型システムを用いる。

これにより、期待される効果は、型付中間言語 MIL が共通計算基盤として、GC、コンパイラ、型システム、グリッド計算などの研究者による新たな研究の展開を促す(例えば、GC 方式の実装を特定の言語の処理系について直接行うのではなく、MIL 処理系に対して行うことで、MIL への翻訳が可能な各種言語について GC 方式を実装することができる)とともに、世界各地のソフトウェアの開発者から利用者への活発な MIL の最新プログラム配布を安全・安心面でサポートする(例えば、グリッド計算などにおける配列アクセス時の冗長な境界チェックの除去は高水準コンパイラが行え、Xi らの DTAL 等の既存研究と同様に除去済みコードの型検査としてサポートし、さらに、Lisp・Scheme 言語などの実装に適した fixnum といった実装技法をサポートすること、であった。

3. 研究の方法

本研究の挑戦的特色は、高水準言語を固定せず、実計算機の特徴を捉えた仮想機械の RISC 命令を設計する形で高速性と汎用性(将来の高水準言語のサポートを含む)を最大限追求しつつ、近年の型理論の成果(依存型、インターセクション型など)を踏まえて、場合依存型の新たな提案や BDD(二分決定グラフ)の利用提案を含む、計算基盤としての安全性を確立する方法を研究するというアプローチにある。

このような従来にないアプローチをとる本研究課題は、さまざまな困難が予想される挑戦的な課題である。

本研究では、型付中間言語 MIL の開発を言語仕様の設計を中心に進めた。言語仕様の設計においては、MIL の役割と型システムを強く意識した上でシンタックス、セマンティクスの設計を進める。また、低水準命令列に関する既存の依存型による型検査技法をベースとした。ここで、言語の基本となる設計方針は以下の通りとした。

- * 局所変数(「擬似レジスタ」)、必ずヒープに生成されるオブジェクト、「軽い」命令、非局所的制御フローを採用する。

- * 局所的制御フロー、非局所的制御フローなど、制御構造を定める。

- * 基本データ型、アドレス(参照)の扱い、オブジェクトの生成方式など、データ構造を定める。

- * 通常の(GCの用語での mutator の意味での)セマンティクスを定める。

- * ごみ集め処理のセマンティクスを定める。(本研究では GC の用語でいう collector の処理は MIL では記述しない。)

MIL の目指すべき役割としては、汎用性の高さ、高性能、処理系として正確なごみ集め、真の末尾再帰のサポートがある。より具体的には以下の目標に対する研究を行った。

MIL の特色のうち、ビットレベル操作においては、fixnum の実現とシフト命令への対応を目標とした。Lisp 系言語などで使われる fixnum などでは、実際のデータが小さな固定整数の場合も参照の場合もある語データを扱えなければならない。

オブジェクトの初期化においては、循環データ構造への対応を目標とした。オブジェクトの初期化は、通常、フィールド毎の型更新(strong update)を伴う初期化として扱うことができるが、循環データ構造を得るには未来において型においても完成するオブジェクトへの参照を、完成前に用いる必要があるという問題があった。

文字列や配列の一般的取り扱いにおいては、特に高速な多次元配列操作への構造的対応を目標とした。DTAL 等の既存研究では一次元配列における冗長な境界チェックの除去が扱われているが、単純には多次元に拡張できない。

また、正確なごみ集めのサポートを目標

としたが、もっとも挑戦的な課題であった。

4. 研究成果

本研究は、仮想機械の RISC 化を目標に汎用性と高速性を持つ型付中間言語の設計に挑んだ。低水準命令列に関する既存の依存型による型検査技法をベースとし、以下の成果（もしくは重要な今後の課題）を得た。

ビットレベル操作においては、fixnum の実現とシフト命令への対応を目標に、場合依存型という新しい概念の提案と二分決定グラフ(BDD)の利用提案ができるような検討を行った。

場合依存型という新しい概念と、BDD (二分決定グラフ) を用いた命令列の妥当性確認の枠組みに関して研究を進めた。Lisp 系言語などで使われる fixnum などのサポートが場合分けを利用して行われるようになることを確認しつつ、応用を拡大することを目指して、特に、シフト命令への対応について考察した。BDD に関してはもともと掛け算などの演算では、その規模がコンパクトでなくなるといった問題があり、必ずしも万能ではないが、単純なシフト演算であっても、ビット変数間の関係を表す BDD は、指数乗のサイズとなる問題がある。当初、BDD を拡張する方向で検討を進めたが、考えられた範囲では論理演算を分配することができずうまくいかなかった。そこで、もともと BDD では、変数順序を管理している点を利用して、その管理方法をシフト命令に合わせて調整するという方法を考案し、検討を進めている。シフト命令は言語処理系の低水準の実装ではよく利用されるため、本研究内容はサポートされる処理系を拡大するのに意義があると考えている。また、依存型などの型システムを使っているが、Curry-Howard 対応はどうなっているのか考察する必要があることが、発表時の質問などから分かった。特に場合依存型について位置づけを明確にする必要がある。

オブジェクトの初期化においては、循環データ構造への対応を目標に、provisional assumptions という新しい概念を提案できるような検討を行った。

provisional assumptions という新しい概念と依存型による命令列の妥当性確認の枠組みに関して研究を進めた。OCaml のような言語では循環参照を持つようなオブジェクトの初期化が可能である。そのような循環データ構造の初期化を許すようなプログラミング言語の実装に設計中の汎用性の高い型付中間言語を用いられるようにするには、オブジェクトの初期化（特に循環参照）のサポートが必要である。また、関数クロージャのための環境を扱う実装レベルで考えると、再帰関数のためにはクロージャは自身をその環境に含める必要がある。本研究では、初期化しようとするオブジェクトのための最終的に得たい仮定についての provisional assumption という一時的な仮定を導入し、こ

れが仮定されている間は、型情報を操作するための型 coerce 命令のみが許されるようにすることで、正当に（型検査可能な形で）循環参照を持つようなオブジェクトの初期化を可能とした。また、余帰納法との関連を考慮する必要があることが、発表時の質問などから分かった。特に、型 coerce 命令以外の命令も許すことができるのかを明確にする必要がある。

文字列や配列の一般的取り扱いにおいては、特に高速な多次元配列操作への構造的対応を目標に、添え字計算に関する新しい述語を提案できるような検討を行った。

例えば、科学技術計算を行うアプリケーションを高水準言語で作成する場合などにおいては、しばしば多次元配列を使用したいことがある。そのような言語から安全な型付中間言語に翻訳する処理系を考える。高速な多次元配列操作のためには、添字計算のコストとともに、実行時境界チェックのコストをできるだけ省きたい。一般に型付中間言語においては、値に関する情報などを型レベルで表現できるような依存型を用いることにより、後続の命令において不要なチェックなどを省く機会が得られる。そこで、依存型で用いるための多次元配列向けの述語を検討した。

また、正確なごみ集めのサポートを目標としたが、もっとも挑戦的な課題であり、検討は進めたものの、正確な手法を提案できるまでには至らなかった。

研究途上であるが、collector が副作用なしでは変数やオブジェクトを走査できない場合があり、そのような場合はある種の型検査エラーとできるようにするべきということなどが今後の課題として判明している。

5. 主な発表論文等

〔学会発表〕(計 3 件)

八杉 昌宏、高速な多次元配列操作のための型検査方式の検討、日本ソフトウェア科学会第 30 回大会 (ポスター発表)、2013 年 9 月 12 日、東京大学本郷キャンパス

Masahiro Yasugi, Typed Construction of Cyclic Data Structures Using Provisional Assumptions, the 10th Asian Symposium on Programming Languages and Systems (APLAS 2012) (Poster Presentation), 2012 年 12 月 12 日、京都市国際交流会館

八杉 昌宏、ビットレベル低水準命令列の BDD に基づく検証技法に関する考察、ディペンダブルシステムワークショップ & シンポジウム (DSW & DSS 2011) (ポスター発表)、2011 年 12 月 13 日、京都工芸繊維大学

6. 研究組織

(1) 研究代表者

八杉 昌宏 (YASUGI, Masahiro)
九州工業大学・大学院情報工学研究院・教授
研究者番号：30273759