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　３０　年　　６　月　１８　日現在

機関番号：　８２６２６
研究種目：　挑戦的萌芽研究
研究期間：　2015～2017
課題番号：　１５Ｋ１２０１３

研究課題名（和文）ビッグデータ処理の形式検証に向けて

研究課題名（英文）Towards formal verification of big data processing

研究代表者
Ａｆｆｅｌｄｔ　Ｒｅｙｎａｌｄ（AFFELDT, Reynald）

国立研究開発法人産業技術総合研究所・情報・人間工学領域・主任研究員

研究者番号：４０４１５６４１

交付決定額（研究期間全体）：（直接経費）　　1,600,000円

研究成果の概要（和文）：インターネットに接続される機器の増加に伴い、蓄積されるデータが爆発的に増大している。これらの大規模なデータを解析し、活用しようという動きが、いわゆる「ビッグデータ」のもとで進んでいる。しかし、ビッグデータ処理に用いられるプログラムの信頼性について、十分に厳密な検討や検証がなされているとは言いがたい。本研究では、「簡潔データ構造」に着目し、そのアルゴリズム及び実装の安全な開発方法を提案し、評価を行った。具体的には、定理証明支援系Coqを用いて、簡潔データ構造の基本的なアルゴリズムを形式化し、その性質を検証してから、実用的なコードを出力できるようにした。

研究成果の概要（英文）：Data accumulate as the result of the increasing number of devices connected to the Internet. These so-called big data are the object of various analyses whose reliability is important. In this project, we focus on succinct data structures as an example of big data. We propose and evaluate an approach for their safe implementation. Concretely, we formalize and verify standard algorithms for succinct data structures using the Coq proof-assistant, and extend the latter so as to be able to extract safe programs that are usable in practice.

研究分野：　形式検証

キーワード：　定理証明支援系　簡潔データ構造　プログラミング言語OCaml　プログラミング言語C　コード生成器

１．研究開始当初の背景

(1) ビッグデータ
インターネットに接続される機器の増加に伴い、蓄積される行動履歴、センサーデータ、文章などが爆発的に増大している。例えば、各種電子店舗におけるレコメンデーションシステムや、センサーネットワークを活用したインフラ監視などが挙げられる。これらの大規模なデータを解析し、活用しようという動きが、いわゆる「ビッグデータ」のもとで進んでいる。

(2) 簡潔データ構造
「ビッグデータ」の高度で高速な解析技術が求められており、その技術の一つとして、圧縮されたデータ構造に関するアルゴリズムが研究されている。簡潔データ構造は代表的な例である：遺伝子情報の解析やモバイル機器上での日本語入力のための辞書情報の格納などに用いられており、ビッグデータの中心になると期待されている。簡潔データ構造とは最小のメモリ量で様々な操作が可能なデータ構造である。簡潔データ構造を扱うライブラリの実装は様々である(例:C++ の SDSL ライブラリ)。しかし、その性質の保証は困難である。アルゴリズムは低レベルのビット列の操作に基づくので、正しさの保証は困難である。また、最小のメモリ量を利用するという性質の保証も困難である。

(3) ビッグデータ処理の信頼性
今後ビッグデータ処理が医療機器やインフラ監視などに用いられるようになれば、その処理方法の信頼性に対する要求がより高まる。しかし、ビッグデータ処理に用いられるプログラムの信頼性については、十分に厳密な検討や検証がなされているとは言いがたい。特に、簡潔データ構造に基づくソフトウェアの正しさを保証するのは困難である。なぜなら、
1. そのようなソフトウェアをテストするためには、 大規模なテストデータを用意する必要があり、完全に網羅的な検査は現実的でない。
2. 簡潔データ構造は、メモリ消費量が理論的な限界まで少なくなるよう設計されていることが多いが、実装がこれを満たしているかどうかをテストで確かめるのは困難である。

(4) 定理証明支援系による形式検証
一方、ソフトウェアの正しさ及び数学の証明の正しさを保証してくれるツールとして、定理証明支援系の研究が欧米を中心に 1970 年代から継続的に行われている。2000 年代からクリティカルな基盤ソフトウェア(コンパイラ、オペレーティングシステムなど)の検証と大規模な数学の証明(四色定理、ケプラー予想など)の形式化が可能となり、現在定理証明支援系の有用性は広く認められるようになった。

２．研究の目的

我々の最終目的はビッグデータを処理するソフトウェアの実装の正しさを保証することである。上記のようにテストによるチェックは困難であり、形式検証が必要とされる。本研究では、大規模なデータに基づくソフトウェアの正しさを高い信頼で保証できるように、
1. 簡潔データ構造に関する古典的なアルゴリズムの形式検証技術を確立し、
2. 次世代のビッグデータ用のアルゴリズムの開発を支えるために、検証済みの現実的なアルゴリズムの開発方法を用意する。
本研究はビッグデータに関する定理証明支援系による形式検証の初めての試みである。

３．研究の方法

本研究では簡潔データ構造の実装の正しさの保証を定理証明支援系を用いて得るようにする。
(1) 定理証明支援系による形式検証まず、定理証明支援系の仕様言語を用いて、古典的なアルゴリズムの記述とその検証を行う。ここで作成されたライブラリは形式仕様として使う。その抽象的なレベルで効率（実行時とメモリ量の使用）の形式検証を行う。
(2) プログラム変換で定理証明支援系を拡張簡潔データ構造の現実的な実装を得るように、定理証明支援系のコード出力機能を拡張する。その際、抽象的な仕様に対して出力するコードの正しさは最大の議論点となる。

４．研究成果

(1) 簡潔データ構造の基礎を形式化
定理証明支援系 Coq を用いて、簡潔データ構造の基礎を形式化し、基礎的なアルゴリズムを検証した。簡潔データ構造のアルゴリズムは２つの基本的なアルゴリズムに基づく：rank アルゴリズムは最小のメモリで、効率的に(constant-time で)ビット列のビットを数えるアルゴリズムであり、select アルゴリズムは効率的にビット列でビットを検索する。本研究では、まず、rank と select アルゴリズムを抽象的に定義し、その性質の証明に取り組み、定理群を開発した。特に、rank アルゴリズムが利用するメモリ量に関する性質に成功した[雑誌論文(2),学会発表(5)]。また、rank と select 関数の性質をライブラリ化し、そのライブラリを用いて、LOUDS(Level Ordered Unary Degree Sequence)木という代表的な簡潔データ構造とその性質を形式化した[学会発表(2)]。

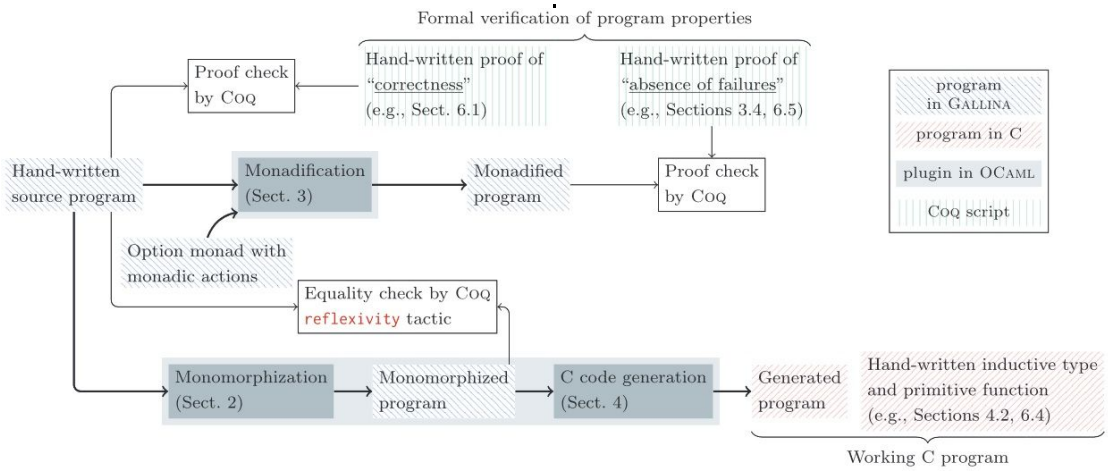

図 1　C コード出力と monadification を合わせた仕組み

(2) 検証済みで実用的な rank 関数の開発検証済みで実用的な rank 関数を得るため、定理証明支援系 Coq を用いた検証方法を提案し、検証実験を行った。簡潔データ構造は低レベルな操作を利用するため、C 言語で記述された通常の rank の実装の検証は困難である。本研究では、抽象的な rank アルゴリズムを検証してから、定理証明支援系 Coq のコード出力機能を用いて、検証済みかつ実行可能なプログラムを形式モデルから得られるようにした。Coq のコード出力によって OCaml 言語の実装を出力できる。しかし、通常のコード出力を使うとビット列は通常のリストデータ構造で出力されるので、期待より多くのメモリ量を必要とする。これにより実用的な簡潔データ構造にならないため、本研究では、効率のいいコードが得られるように、ビット列を表現する OCaml ライブラリを新しく構築した。特に、このライブラリは最新の Intel アーキテクチャの命令を利用することができるので、パフォーマンスの高い実装を得られた[雑誌論文(2),学会発表(5)]。

(3) Coq のコード出力
上記で説明した検証済みで実用的な rank 関数は主に関数型言語 OCaml で記述されているため、効率の改善の余地がある。さらに効率の良いプログラムを得られるように、Coq のコード出力の新機能を提案した。その機能によって、形式モデルから直接効率の良い低レベルの C 言語のプログラムを得られる。ただし、その新コード出力機能の信頼性の保証は困難である。Coq が関数型プログラミング言語であるため、命令型の C 言語への出力は自明ではない。従って、提案したプログラム変換の一部は Coq が保証できるように構成した。具体的には、プログラム変換の最初のステップは（monomorphization と A-normal form 変換という）Coq の型検査が確認できるようにコード出力機能をデザインした。そのコード生成の手法を様々な実用的な応用に適用した。特に、上記のように形式検証した rank アルゴリズムに適用した。その結果として、C 言語のコードで記述された実用的な実装を得られた[雑誌論文(1)]。更に、コード生成器を拡張した。具体的には、C 言語で記述された実装の効率改善のため、linearity 解析という最適化を実装した[学会発表(2)]。

(4) monadification による安全な C コードの出力
上記の説明のように C コード出力機能の安全性は重要である。そのため、プログラム変換の一部は Coq が検証できるように monomorphization を使用した。更に、C コード出力機能の信頼性の向上のため、monadification という概念を導入した。monadification によって、Coq のレベルで使用する抽象的なデータ構造（例えば、自然数）と C 言語で使用するデータ構造（例えば、コンピュータによる整数）の変換の安全性を確かめられる。その正しさは monad という概念を利用した証明を可能とした。C コード出力と同様、monadification を Coq プラグインとして実装した、応用した。また、monadification の応用の幅は広く、正しい値を返すという以外の性質（計算量など）も形式的に検証できた。
C コード出力と monadification を組み合わせると最終的に、形式的に安全性を確かめられる生成器ができ(図 1)[雑誌論文(1)，学会発表(3)]、これをオープンソースコードとして配布した。

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 2 件）

(1) Akira Tanaka, Reynald Affeldt, and Jacques Garrigue, Safe low-level code generation in Coq using monomorphization and monadification, Journal of Information Processing, 査読有, 26 巻, 2018, 54-72
DOI: 10.2197/ipsjjip.26.54

(2) Akira Tanaka, Reynald Affeldt, and Jacques Garrigue, Formal Verification of the rank Algorithm for Succinct Data Structures, 18th International Conference on Formal Engineering Methods (ICFEM 2016), Tokyo, Japan, November 14-18, 2016, Lecture Notes in Computer Science, 査読有, 10009 巻, 2016, 243-260
DOI: 10.1007/978-3-319-47846-3_16

〔学会発表〕(計 5 件)

(1) Reynald Affeldt, Generation of Code from Coq for Succinct Data Structures, ENabling TRust through 0s Proofs... and beYond (Entropy 2018), Villeneuve d'Ascq, France, 2018/01/26.

(2) Akira Tanaka, Reynald Affeldt and Jacques Garrigue, Future Work Towards a Coq Library of Succinct Data Structures, 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(PPL2018), 鳥取県米子市, 2018/03/06, ポスター

(3) Akira Tanaka, Reynald Affeldt and Jacques Garrigue, Safe Low-level Code Generation in Coq using Monomorphization and Monadification, 第 114 回プログラミング研究発表会 情報処理学会プログラミング研究会, 静岡市, 2017/06/09

(4) Akira Tanaka, Reynald Affeldt and Jacques Garrigue, Certified Mon{omorphiz|adific}ation of Gallina for Low-level Code Extraction, 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 (PPL2017), 山梨県笛吹市, 2017/03/09, ポスター

(5) Akira Tanaka, Reynald Affeldt, and Jacques Garrigue, Formal Verification of the rank Function for Succinct Data Structures, 第 18 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 (PPL2016), 岡山県玉野市, March 7-9, 2016, 論文賞

〔図書〕(計 0 件)

〔産業財産権〕

○出願状況 (計 0 件)

○取得状況 (計 0 件)

〔その他〕
プロジェクト ホームページ：
https://staff.aist.go.jp/tanaka-akira/succinct

C コード出力プラグイン：
https://github.com/akr/codegen

monadification プラグイン：
https://github.com/akr/monadification

６．研究組織

(1) 研究代表者
アフエルト　レナルド (AFFELDT Reynald)
国立研究開発法人産業技術総合研究所・情報・人間工学領域・主任研究員
研究者番号: 40415641

(2) 研究分担者
田中　哲 (TANAKA Akira)
国立研究開発法人産業技術総合研究所・情報・人間工学領域・主任研究員
研究者番号: 10357452

ガリグ ジャック (GARRIGUE Jacques)
名古屋大学・多元数理科学研究科・准教授
研究分担者: 80273530

(3)連携研究者
N.A.
(4)研究協力者
N.A.