

科学研究費助成事業 研究成果報告書

平成 29 年 5 月 23 日現在

機関番号：32503

研究種目：若手研究(B)

研究期間：2013～2016

課題番号：25871113

研究課題名(和文) PGAS言語のメモリー貫性に関するプログラム検証理論とその実装

研究課題名(英文) A Program Verification Theory of Memory Consistency Models on PGAS languages and its Implementation

研究代表者

安部 達也 (Abe, Tatsuya)

千葉工業大学・人工知能・ソフトウェア技術研究センター・上席研究員

研究者番号：50547388

交付決定額(研究期間全体)：(直接経費) 3,300,000円

研究成果の概要(和文)：PGAS言語で採用されているメモリー貫性モデルを調査し、それらを統一的に扱うプログラム検証理論の一つを提案し、ユーザ定義可能なメモリー貫性モデルまでも考慮することができるモデル検査を行うための実装McSPINを開発・公開した。いくつかの最適化を提案し、それに関する論文を雑誌・国際会議に投稿し、採択された。プログラム論理の構築にも成功し、これを記載した論文を雑誌に投稿し、採択された。また、検証理論で使用する記述言語で、実際のPGAS言語であるところのXcalableMPのメモリー貫性モデルを記述した。これはXcalableMPの仕様策定の規格部会で採用が承認され1.3版から収録される予定である。

研究成果の概要(英文)：I have investigated various memory consistency models (MCMs) which are adopted by PGAS languages. I have proposed a program verification theory to handle the MCMs, and developed and released McSPIN, a model checker with user-defined MCMs. Through the development, I have proposed some optimizations of model checking with MCMs. Six papers about them have been published. I have also constructed a program logic to handle MCMs, and one paper about the logic has been published. I have formally written the MCM which is adopted by a PGAS language XcalableMP in the description language of the program verification theory that I have proposed. The formal description of the MCM has been accepted by the XcalableMP working group, and will be published in the XcalableMP specification Version 1.3.

研究分野：ソフトウェア

キーワード：メモリー貫性モデル プログラム検証 モデル検査 プログラム論理

1. 研究開始当初の背景

プログラムは、主に計算機の計算性能(実行時間等)の向上のためそのコードに記述された順番どおりに実行されていると仮定できません。また、全てのスレッドから同じように観測されるとも仮定できません。このため、その順番とその順番の見え方を規定する、いわゆるメモリー貫性モデルは仕様に明確に記載されている必要があります。しかし、様々なアーキテクチャの違いを吸収しつつ、また、プログラム実行順序(プログラミングの容易さ)と実装の柔軟性とのトレードオフを考慮するメモリー貫性モデルを与えることは容易ではありません。(極端なメモリー貫性モデルを選択しない限りは)プログラムは記述された通りに実行されるものでもなければ、それが全ての観測者から同じように観測されるとは限らないからです。まして、並列プログラムにおいては、プログラムを実行する計算機も複数であるため、より困難です。また、並列分散計算環境を利用するための並列プログラミング言語として対象をPGASに絞って見たところでも、PGASには Unified Parallel C (CAF)、Coarray Fortran (CAF)、XcalableMP (XMP) 等、複数存在し、それぞれ異なるメモリー貫性モデルを持ちます。プログラマは言語ごとにその異なるメモリー貫性モデルの下でのプログラミングを強いられます。

このような困難があるにも関わらず計算機科学の研究者・開発者の不断の努力により C、Java 等の逐次プログラミング言語の仕様にはメモリー貫性モデルがそれぞれ与えられてきました。そして現在、並列分散計算環境が一般的になってきており、並列プログラミング言語にメモリー貫性モデルを与えることの重要性が高まっています。並列プログラミング言語にその並列性を何に利用するかによりいくつかの種類があるが、本研究では並列性を高性能計算のために利用することとし、そのための並列プログラミング言語である PGAS を扱います。

2. 研究の目的

以上の背景と動機から、PGAS 全体に渡るメモリー貫性モデルに関する考察とその下での PGAS プログラムの挙動に関する検査器は必須です。本研究では PGAS 全体に渡るメモリー貫性モデルの形式化を行います。形式化を行った後、各 PGAS に対しメモリー貫性モデルをとることができるツールを実装することで、メモリー貫性モデルに対して可変なツールを得ます。

次に、そのツールをプログラムを入力にとることができるプログラム検証器に拡張することで、メモリー貫性モデルの変更に柔軟なプログラム検証器の開発を行います。この拡張には、現在代表者が開発しているツール

XMP-SPIN との結合により行います。このツールは CAF や XMP のプログラムを入力として Promela (モデル検査器 SPIN の入力言語) のプログラムを出力とします。この意味で、本研究は代表者が開発しているツールで組み込みとされているメモリー貫性モデルをモジュール化する研究であるともいえます。

3. 研究の方法

複数の計算ノード上に存在する複数のメモリーに対する統一的なアクセス方法を提供する、いわゆる PGAS 言語である UPC と CAF のメモリー貫性モデルの調査を行いました。さらに、コンピュータアーキテクチャである Itanium のメモリー貫性モデルの調査も行いました。その上で、少なくともこれら三つのメモリー貫性モデルを記述可能にする汎用的なモデル(以下、ベースモデルと呼ぶ)とその上でこれら三つのメモリー貫性モデルを記述するための形式言語の定義を与えました。この定義の下で、この形式言語で記述されたメモリー貫性モデルは、ベースモデル上におけるプログラムの実行トレースのうち、どの実行トレースを許容可能とするかという規則と見なされます。これらの定義を与えたことにより、曖昧性や多義性を含みがちな自然言語によって記述されるメモリー貫性モデルを形式的に扱えるようにしました。

4. 研究成果

メモリー貫性モデルを形式的に扱うことができる利点の一つとして、形式化されたメモリー貫性モデルを入力にとるモデル検査器 McSPIN の試作を前述の XMP-SPIN を大幅に拡張することにより行いました。このモデル検査器で UPC Language Specifications Version 1.2 の Appendix B に載っているサンプルプログラムと Itanium の仕様書 (A Formal Specification of Intel Itanium Processor Family Memory Ordering) の Appendix A に載っているサンプルプログラムに対し検査を行い、検査器の正しさ・有用性・汎用性を確認しました。[6,7,8]

また、この評価を通して(研究費申請時には気づいていなかった)モデル検査をより効率的に行うことができる方法、論理式として形式化されたメモリー貫性モデルの形式から抽出可能な述語抽象化を発見したため、その方法に関しても論文を執筆しました。[5]

さらに、検査の最適化に関して、これもまた同様に形式化されたメモリー貫性モデルの形式を考慮することで、検証したい性質に関して統合が可能な状態の発見手法を提案・開発しました。これにより、検査しなければならぬ状態の数の削減に成功しました。[3]

検査器 McSPIN の開発を通して得た知見により、メモリー貫性モデルを考慮したプログラム検証において正しさを保証するためのプログラム論理の構築にも成功しました。[1] この論理では、データレースを含むプログラムの検査も可能であることにより他の既存研究と一線を画しています。

本研究をまとめあげた論文の執筆も行いました。概論を国際ワークショップ The 6th Workshop on Syntax and Semantics of Low-Level Languages のショートペーパーで紹介しました。[4] また、総論を International Journal on Software Tools for Technology Transfer にフルペーパーで執筆しました。[2]

理化学研究所・筑波大学が中心になって開発している PGAS 言語である XcalableMP の仕様書へ本研究課題で行った形式化されたメモリー貫性モデルによる仕様記述方法を採用し Appendix E の執筆を行いました。2017 年 1 月 12 日に行われた XcalableMP の仕様策定の規格部会で承認され、Version 1.3 から採用される予定です。

また、主にアメリカで開発されている並列プログラミング言語 Chapel の開発者から仕様書に掲載しているプログラム例がメモリー貫性モデルの観点から正しいかの確認を Cray 社の Michael Ferguson 博士から依頼されたため、これのフィードバックを行いました。

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. Tatsuya Abe and Toshiyuki Maeda. Concurrent program logic for relaxed memory consistency models with dependencies across loop iterations. Journal of Information Processing, 25:244-255, 2017. DOI:10.2197/ipsjip.25.244 査読あり

2. Tatsuya Abe and Toshiyuki Maeda. A general model checking framework for various memory consistency models. International Journal on Software Tools for Technology Transfer, 2016. DOI:10.1007/s10009-016-0429-y 査読あり

3. Tatsuya Abe, Tomoharu Ugawa, Toshiyuki Maeda, and Kousuke Matsumoto. Reducing state explosion for software model checking with relaxed memory consistency models. In Proceedings of the 2nd International Symposium on Dependable

Software Engineering: Theories, Tools and Applications (SETTA), volume 9984 of Lecture Notes in Computer Science, pages 118--135. Beijing, China, November 9--11, 2016. DOI:10.1007/978-3-319-47677-3_8 査読あり

4. Tatsuya Abe and Toshiyuki Maeda. Towards a unified verification theory for various memory consistency models. The 6th Workshop on Syntax and Semantics of Low-Level Languages (LOLA), online 2 pages. Kyoto, Japan, July 5, 2015. 査読あり

5. Tatsuya Abe and Toshiyuki Maeda. Optimization of a general model checking framework for various memory consistency models. In Proceedings of the 8th Conference on Partitioned Global Address Space Programming Models (PGAS), No. 14, online 10 pages. Eugene, United States, October 7--10, 2014. DOI:10.1145/2676870.2676878 査読あり

6. Tatsuya Abe and Toshiyuki Maeda. A general model checking framework for various memory consistency models. In Proceedings of the 19th Workshop on High-Level Parallel Programming Models and Supportive Environments (HIPS), pages 332--341. Phoenix, United States, May 19--23, 2014. DOI:10.1109/IPDPSW.2014.47 査読あり

7. Tatsuya Abe and Toshiyuki Maeda. Model checking with user-definable memory consistency models. In Proceedings of the 7th Conference on Partitioned Global Address Space Programming Models (PGAS), pages 225--230. Edinburgh, United Kingdom, October 3--4, 2013. 査読あり

8. Tatsuya Abe, Toshiyuki Maeda, and Mitsuhiro Sato. Model checking stencil computations written in a partitioned global address space language. In Proceedings of the 18th Workshop on High-Level Parallel Programming Models and Supportive Environments (HIPS), pages 365--374. Cambridge, United States, May 20--24, 2013. DOI:10.1109/IPDPSW.2013.90 査読あり

[学会発表](計 0 件)

[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]

ホームページ等

McSPIN: Model Checker with Memory Models

<https://bitbucket.org/abet/mcspin/>

6. 研究組織

(1) 研究代表者

安部 達也 (ABE, Tatsuya)

千葉工業大学・人工知能・ソフトウェア技

術研究センター・上席研究員

研究者番号：50547388