様式 C -19

科学研究費補助金研究成果報告書

平成　21 年　6 月　4 日現在

研究種目：若手研究（B）
研究期間：2006～2008
課題番号：18700021
研究課題名（和文）　構成的アルゴリズム論に基づくタスク並列処理の理論とその実用化に関する研究
研究課題名（英文）　Theory and Practice of Task-Parallel Computation Based on Constructive Algorithmics

研究代表者
松崎　公紀（MATSUZAKI KIMINORI）
東京大学・大学院情報理工学系研究科・助教
研究者番号：30401243

研究成果の概要：スケルトン並列プログラミングは，並列プログラムを容易に作成できるようにするための手法のひとつである．本研究では，ある計算をその計算が操作するデータ構造の観点から系統的に定式化する構成的アルゴリズム論に基づいて，スケルトン並列プログラミングの理論と実現に関して研究を進めた．木構造データの効率的な並列処理のための新しいデータ構造，スケジューリングアルゴリズム，ライブラリの実現などの成果を得た．

交付額

（金額単位：円）

	直接経費	間接経費	合　計
2006 年度	1,100,000	0	1,100,000
2007 年度	800,000	0	800,000
2008 年度	800,000	240,000	1,040,000
年度			
年度			
総　計	2,700,000	240,000	2,940,000

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：プログラム言語，ハイパフォーマンス・コンピューティング，アルゴリズム，情報基礎，スケルトン並列プログラミング

１．研究開始当初の背景

ひとつの CPU や計算機では扱えないような大きな問題を解くにあたって，並列計算は非常に重要な手法である．この並列計算を行うためのハードウェアである並列計算機は，PC やネットワークの高速化や低価格化によって，比較的容易に手に入れることができるようになってきた．また，それまでの CPU の機能を持つ計算コアを複数持つようなマルチコア CPU も出現し，計算機を効率良く利用するためには並列プログラムを作成することが重要であることが広く認知されはじめた．

一方で，そのような効率の良い並列プログラムを作成することは，逐次プログラムを作成することと比較して難しい．そもそもアルゴリズムが並列に計算できる必要があること，複数の計算資源に対して処理を適切に分散させる必要があること，同期やプロセッサ間通信を行う必要があることなどがその理由である．したがって，これらの並列プログラムに特有の問題点を解決するための手法が求められている．

スケルトン並列プログラミング（Skeletal

Parallel Programming）は，そのための有望な手法のひとつである．スケルトン並列プログラミングでは，「並列スケルトン」と呼ばれる，並列計算に良く出現する計算パターンを抽象化したブロックを組み合わせることで並列プログラムを作成する．並列性が並列スケルトンの内部に隠蔽されるため，ユーザは逐次プログラムを作成するのと同じようにして並列プログラムを作成できる．

スケルトン並列プログラミングの中でも，データ並列処理を並列スケルトンによって実現する手法に関して多くの研究が行われてきている．特に研究代表者が所属するグループでは，「構成的アルゴリズム論」と呼ばれる理論に基づく系統的なデータ並列スケルトンに関する研究を行ってきた．構成的アルゴリズム論は，計算が操作するデータ構造に基づいてその計算を系統的に捉えるための理論であり，それまでにも逐次のプログラミングを対象としてさまざまな成果をあげてきた．並列計算に関しては，これまでに，規則的なデータ構造であるリスト（一次元配列），行列（二次元配列），木に対する並列スケルトンの定式化とその実現が行われていた．

２．研究の目的

スケルトン並列プログラミングに関して多くの研究が行われているが，残念ながらその多くはアドホックに並列スケルトンを定義しており，理論的な定式化などが不十分であった．この理論的な定式化は，並列スケルトンを用いたプログラムの効率を改善する最適化を行う上で重要になるものである．一方，「構成的アルゴリズム論」に基づく並列スケルトンの定式化は理論的には良いものの，規則的ではないデータ構造やデータ構造の動的な変化に対応することはできていなかった．これらの問題点は，スケルトン並列プログラミングの実用化における大きな問題点であった．

本研究では，タスク並列計算の観点から，スケルトン並列プログラミングに対する理論を構築し，また実用的なスケルトン並列プログラミングライブラリを作成することを目的とする．

具体的には，次の 3 つが主な研究対象となる．

(1) 研究代表者がそれまで取り組んできたデータ構造「木」に対する並列スケルトンに関して，扱うデータの動的な変化に対応できるような拡張を与える．
(2) タスク構造（多くの場合には，木構造や有向非循環グラフ）に対して，そのタスク構造の特徴に応じた良いスケジューリング手法を与える．
(3) 上記の理論に基づいて，より実用的なスケルトン並列プログラミングライブラリを構築する．

３．研究の方法

研究の目的において述べた研究対象について，さらに小項目に分けながら，理論的な研究と実践的な研究を並行して進めた．

理論的な研究については，基本的なアイデアを確認した後に，当該分野に強く関係する国内外の研究会や国際会議において，積極的に議論を行った．例えば，研究対象の(1)については，アイデアをまとめた資料を作成し（最終的には，テクニカルレポート[2]としてまとめた）国際的な研究者と議論し，また国内の研究会にて発表を行った[17]．また，研究対象の(2)については，当該分野にて世界をリードしているグループを訪問し，共同研究という形で研究をスタートさせた．

(3)のスケルトン並列プログラミングライブラリの構築については，それまでにも研究代表者のグループにて開発してきたライブラリを拡張することで行った．また，開発した並列スケルトンライブラリは，ウェブ上にて公開し広く利用・議論ができるようにした．

４．研究成果

本研究の研究成果は，大きく次の 2 点にまとめることができる．

(1) 木構造上での並列計算のための新しい定式化
(2) 並列スケルトンライブラリの構築

それまでにも木構造データに対するスケルトン並列プログラミングに関する研究が行われていたが，それらのほとんどは動的に形が変わることがない二分木という制限された木構造に対するものであった．本研究では，より一般の木構造に対してスケルトン並列プログラミングを行うことができるように，その理論の研究を行った．この成果のひとつは，子の数に制限のないデータ構造（薔薇木と呼ばれる）に対するスケルトン並列プログラミング手法の提案である．一般の木構造に対して，それらを操作する並列スケルトンを定式化し，それらを用いて並列プログラムを作成するための手法を与えた．この成果は，論文誌 Parallel Computing にて発表されている[11]．木構造に対する研究成果のもうひとつは，動的にその形が変化するような木構造に対する並列計算の定式化である．特に，本研究では，データ構造の観点から定式化を行い，「平衡三分木」と呼ぶ新しい並列計算のためのデータ構造を提案した（図 1）．

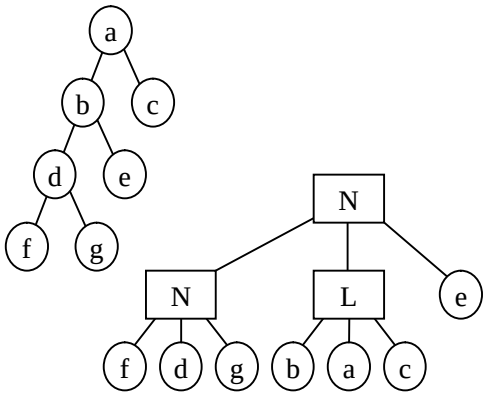

図 1. 二分木（左上）とその平衡三分木表現

このデータ構造を用いることにより，実行時に動的に形が変化するような木構造に対して，効率の保証された並列計算が可能となる．また，この新しいデータ構造における並列計算の導出法に関しても提案を行った．これらの成果は，テクニカルレポート 2 件[2, 3]，国内外の会議 [1, 16, 17] にて発表した．

また，それまでに研究代表者のグループで開発してきた並列スケルトンライブラリ「助っ人」の実装を改良した．「助っ人」は，C++ と MPI によって実装された並列スケルトンライブラリであり，それまでに研究されてきた理論に基づく並列スケルトンを提供するものである．本研究では，複数の並列スケルトンを使った場合の最適化機構の改良を行った．並列スケルトンを用いたプログラムは，ユーザが並列性や効率を意識して書いたプログラムに比較して効率が悪いことがあり，その最適化を行うことは重要な課題である．本研究では，関数プログラミングの分野で研究されてきた融合変換による最適化機構を，C++のテンプレート機能を利用することで実現した．また，並列スケルトンのインターフェイスを改良し，使い勝手と効率を向上させた．この新しい並列スケルトンライブラリの実装方法については，[15]にて発表した．

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 14 件）

[1] A. Morihata, K. Matsuzaki, Z. Hu, and M. Takeichi. The third homomorphism theorem on trees: upwards & downwards leads to divide-and-conquer. In *Proceedings of the 36th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Language (POPL2009)*, pp. 177-185, 2009.（査読有）

[2] K. Matsuzaki and A. Morihata. Ternary-tree representation of binary trees and balancing algorithms. *Technical Report (METR), Department of Mathematical Informatics, The University of Tokyo*, METR2008-30, pp. 1-18, 2008.（査読無）

[3] A. Morihata and K. Matsuzaki. A parallel tree contraction algorithm on non-binary trees. *Technical Report (METR), Department of Mathematical Informatics, The University of Tokyo*, METR2008-27, pp. 1-7, 2008.（査読無）

[6] 松崎公紀，胡振江，武市正人．リスト上の最大マーク付け問題を解く並列プログラムの導出．**情報処理学会論文誌：プログラミング**, Vol. 49 (SIG3), pp. 16-27, 2008.（査読有）

[4] K. Emoto, K. Matsuzaki, Z. Hu, and M. Takeichi. Domain-specific optimization strategy for skeleton programs. In *Proceedings of International Conference on Parallel and Distributed Computing (EuroPar2007)*, pp. 705-714, 2007.（査読有）

[7] 野村 芳明，江本 健斗，松崎 公紀，胡振江，武市 正人．木スケルトンによる XPath クエリの並列化とその評価．**コンピュータソフトウェア**, Vol. 24(3), pp. 51-62, 2007.（査読有）

[5] K. Emoto, K. Matsuzaki, Z. Hu, and M. Takeichi. Domain-specific optimization for skeleton programs involving neighbor elements. *Technical Report (METR), Department of Mathematical Informatics, The University of Tokyo*, METR2007-05, pp. 1-29, 2007.（査読無）

[8] K. Morita, A. Morihata, K. Matsuzaki, Z. Hu, and M. Takeichi. Automatic inversion generates divide-and-conquer parallel programs. In *Proceedings of the ACM SIGPLAN 2007 Conference on Programming Language Design and Implementation (PLDI2007)*, pp. 146-155, 2007.（査読有）

[9] K. Matsuzaki. Efficient implementation of tree accumulations on distributed-memory parallel computers. In part of *The International Conference on Computational Science (ICCS 2007)*, pp. 609-616, 2007.（査読有）

[10] K. Kakehi, K. Matsuzaki, and K. Emoto. Efficient parallel tree reductions on distributed memory environments. In part of *The International Conference on Computational Science (ICCS 2007)*, pp. 601-608, 2007.（査読有）

[11] K. Matsuzaki, Z. Hu, and M. Takeichi. Parallel skeletons for manipulating general trees. *Parallel Computing*, Vol. 32(7-8), pp. 590-603, 2006.（査読有）

[12] K. Matsuzaki, K. Emoto, H. Iwasaki, and Z. Hu. A library of constructive skeletons for sequential style of parallel programming (invited paper). In *Proceedings of 1st International Conference on Scalable Information Systems (Infoscale2006)*, 2006.（査読無）

[13] K. Emoto, K. Matsuzaki, Z. Hu, and M. Takeichi. Surrounding theorem: developing parallel programs for matrix-convolutions. In *Proceedings of International Conference on Parallel Computing (EuroPar2006)*, pp. 605-614, 2006.（査読有）

[14] K. Matsuzaki, Z. Hu, and M. Takeichi. Towards automatic parallelization of tree reductions in dynamic programming. In *Proceedings of 18th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA2006)*, pp. 39-48, 2006.（査読有）

〔学会発表〕（計 7 件）

[15] 松崎 公紀, 江本 健斗. 並列スケルトンライブラリ「助っ人」の実現. **第 50 回プログラミングシンポジウム**, 箱根, 神奈川県, 2009.1.13.

[16] K. Matsuzaki. Associativity for parallel tree computation. *The 3rd DIKU-IST Joint Workshop on Foundation of Software*, Denmark, 2007.10.5.

[17] 松崎公紀, 森畑明昌, 胡振江, 武市正人. Associativity for parallel tree computation. **日本ソフトウェア科学会第 24 回大会**, 奈良先端科学技術大学院大学, 奈良県, 2007.9.13.

[18] 江本健斗, 松崎公紀, 胡振江, 武市正人. 近傍要素を必要とするスケルトンプログラムの最適化. **第 9 回プログラムおよびプログラミング言語ワークショップ (PPL2007)**, 加賀温泉, 石川県, 2007.3.9.

[19] 森田和孝, 森畑明昌, 松崎公紀, 胡振江, 武市正人. Automatic inversion generates divide-and-conquer parallel programs. **第 9 回プログラムおよびプログラミング言語ワークショップ (PPL2007)**, 加賀温泉, 石川県, 2007.3.9.

[20] 松崎公紀, 胡振江, 武市正人. Towards automatic parallelization of tree reductions in dynamic programming. **第 9 回プログラムおよびプログラミング言語ワークショップ (PPL2007)**, 加賀温泉, 石川県, 2007.3.9.

[21] K. Matsuzaki and N. Ohkawa. A parallelization tool for tree reductions. *The 2nd DIKU-IST Joint Workshop on Foundation of Software*, 葉山, 神奈川県, 2006.4.22.

〔その他〕
研究代表者のホームページ
http://www.ipl.t.u-tokyo.ac.jp/~kmatsu/

並列スケルトンライブラリ「助っ人」のホームページ
http://www.ipl.t.u-tokyo.ac.jp/sketo/

６．研究組織
(1)研究代表者
松崎　公紀（MATSUZAKI KIMINORI）
東京大学・大学院情報理工学系研究科・助教
研究者番号：30401243

(2)研究分担者

(3)連携研究者