

令和 5 年 6 月 16 日現在

機関番号：26402

研究種目：基盤研究(C)（一般）

研究期間：2017～2022

課題番号：17K00108

研究課題名（和文）並列分散XMLクエリアルゴリズムおよびその断片化適応型ライブラリの実現

研究課題名（英文）Design and implementation of fragmentation-adaptive parallel and distributed XML query library

研究代表者

松崎 公紀（MATSUZAKI, Kiminori）

高知工科大学・情報学群・教授

研究者番号：30401243

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）：本研究は、これまでに研究代表者が得た並列化手法を適用することで、データ断片化手法により分散されたXMLデータに適用できる並列クエリアルゴリズムを提案・評価することを目的としたものである。本研究における最も重要な研究成果は、最先端のXMLデータベースBaseX上でXPathステップ毎の並列化を実装し評価を行ったことである。また、XPathクエリを一部書き換えた上で木全体に対して並列にクエリを実行する、クエリ分割型の並列化手法についても性能評価を行った。

研究成果の学術的意義や社会的意義

様々なデータが（従来の表形式のデータベースではなく）構造の緩やかな形式で保存・活用されており、100GBを越えるような大規模データも存在する。本研究成果は、そのような大規模木構造データに対するクエリ処理をより高速に実行するための基礎を与えるものである。とくに、最新のデータベース上で並列クエリを実装できることを示したことにより、全てを独自に実装するよりも現実的な実装が可能であることを示した。

研究成果の概要（英文）：We aimed to develop and evaluate parallel query algorithms for distributed XML data, based on the parallelization methods developed by the PI. The main result in this project was that we implemented and evaluated XPath step-wise parallelization algorithms implemented on top of the state-of-the-art XML database, BaseX. We also evaluated query-partitioning parallelization technique, which executes queries on the whole tree in parallel.

研究分野：並列プログラミング手法

キーワード：XML 並列分散処理 データベース MapReduce

## 1. 研究開始当初の背景

様々なデータが(従来の表形式のデータベースではなく)構造の緩やかな(semi-structure)形式で保存・活用されている。それらは特に木構造データを表現するXMLやJSONなどの形式をとることが多い。XMLなどの形式で保存・提供されるデータは大規模化しており、研究代表者が知る範囲でも数100GBを越えるデータが存在している(例:蛋白質DBであるUniProtKB)。そのような大規模XMLデータを効率良く操作するためには、複数の計算機を用いた並列処理が必要となる。XMLなどで表される木構造データに対するクエリ処理ではXPathやXQueryが広く用いられている。それらXPathやXQueryにおける木構造に対するクエリ処理の中心はNavigational XPathと呼ばれる。Navigational XPathクエリには、親→子方向の走査だけでなく、子→親方向の走査、兄弟間の(順序を考慮した)走査、追加の条件を指定する述語が含まれる。Navigational XPath全体を効率良く実行できる並列分散処理系は存在していなかった。

XPathやXQuery自体は新しくないものの、それらの並列実行に関する研究は、2009年のBordawekarらの研究[1]以降活発に行われてきた。しかしながら、それら並列クエリ実行に関する研究の多くは共有メモリ環境における並列化に焦点をあてており、分散メモリ環境における並列化に対応したアルゴリズムや実装に関する研究は少なかった。並列分散実行の基礎となるXMLデータの断片化(XML fragmentation)手法については、複数の手法が提案されていた[2]。

## 2. 研究の目的

本研究では、これまでに研究代表者が得た並列化手法を適用することで、データ断片化手法により分散されたXMLデータに適用できる並列クエリアルゴリズムを提案・評価することを目的とする。

具体的には、以下の3つの断片化手法とそれに対する並列クエリアルゴリズムを研究の対象として設定した。

- 水平方向の断片化(horizontal fragmentation):比較的規則的でフラットな木構造データに対して有効な断片化手法である。水平方向の断片化により分散されたデータに対しては、XPathステップ毎の並列化手法[3]が有効である。
- 垂直方向の断片化(vertical fragmentation):複数の木データを結合(join)したような木構造データについて、その結合位置で断片化するものである。垂直方向の断片化により分散されたデータに対しては、(木)オートマトンに基づく並列化手法[4]が適用できる。
- m-bridgeに基づく断片化:この断片化手法は、木の構造に依らず任意の木に対して適用できることが大きな利点である。m-bridgeに基づく断片化により分散されたデータに対しては、(木)オートマトンに基づく並列化手法[4]が適用できる。

## 3. 研究の方法

本研究の中心的技術は、「XPathステップ毎の並列化」と「(木)オートマトンに基づく並列化」の2つの手法によるNavigational XPathクエリの並列評価である。これらの2つの手法について、既存の研究成果をさらに拡張してNavigational XPathクエリを効率良く実行できるアルゴリズムを提案し、それぞれに対応して断片化されたデータに対する評価を行う。

本研究は、研究代表者が中心となり実施するものである。その一部であるXPathステップ毎の並列化アルゴリズムとその拡張、一部のライブラリ実装については、研究代表者が指導する大学院博士課程学生(W. Hao氏、~平成30年度)と研究を実施した。また、XMLデータベースBaseX上でのXPathクエリ実装については、木スケルトンの汎用的実装について共同研究を行ってきた佐藤重幸氏(高知工科大学ポスドク研究員、~平成30年度)と共同で研究を実施した。

本研究における実験は、研究代表者が所有する2CPU(12コア)の共有メモリサーバ、もしくは、所属する高知工科大学の共用機器である大規模GPUクラスタを利用して行う。

(1) XPathステップ毎の並列化:研究代表者らが提案したXPathステップ毎に並列化を行うアルゴリズム[3]を拡張し、Navigational XPathクエリに含まれる順序に依存する走査や述語を扱えるようにする。

(2) (木) オートマトンに基づく並列化: 研究代表者らは, オートマトンに基づくアルゴリズムを用いて XPath の基本的なサブセットを計算できることを示した [4]. より複雑な述語への対応に主眼を置きアルゴリズムの拡張と性能評価を行うことを, 当初は研究課題として考えていた.

(3) MapReduce 向け木分割処理の実装: m-bridge に基づく木断片化は, 並列計算に対して良い性質を持つものの, 分割そのものは逐次的に行われていた. 数 100 GB クラスのデータに対応するために, この分割そのものも効率良く並列処理する方法を提案する.

(4) XQuery サブセットへの拡張: 上記 (1) と (2) の並列化アルゴリズムを拡張することで, より記述力の高い XQuery のための並列化アルゴリズムを提案することを, 当初は研究課題として考えていた.

(5) 深層強化学習により生成されるデータの保存・利用のためのライブラリ: 近年, 深層強化学習が様々な応用分野で利用されている. そこで生成される大規模データは学習の途中で捨てられることも多いが, 評価・追試のためには適切に保存されるべきである. そのような大規模データを並列分散処理により小さなオーバーヘッドで保存するためのライブラリを開発する.

#### 4. 研究成果

「研究の方法」において示した各研究課題に対し, 得られた研究成果をそれぞれ以下に示す.

(1) 研究代表者らの文献 [3] では, 木構造データを Partial Trees と呼ぶ独自の形で持つ独自実装を行っていた. これに対し, 木構造データを直接保存・操作できる XML データベースの上で実装できれば性能や拡張性の上で優れるため, 最先端の XML データベース BaseX 上で XPath ステップ毎の並列化を実装し評価を行った. 本成果は, 国際会議 ADBIS2018 にて発表した (さらに, 完全版の論文を arXiv にて公開した). また, XPath クエリを一部書き換えた上で, 木全体に対して並列にクエリを実行するクエリ分割型の並列化手法についても性能評価を行い, 情報処理学会全国大会にて発表した. さらに, 文献[3]の独自実装を改良して性能向上させた結果については国際会議 CloudComp2021 にて発表した.

(2) 上記 (1) の研究結果より, とくに XML データベースを用いて効率良く木構造を分割・分散できる範囲においては, 直接 XPath ステップを並列化することで大きなオーバーヘッドなくクエリを並列実行できることが分かった. 一方, オートマトンに基づく並列化は, その手法に由来するオーバーヘッドが避けられない. このことから, 本研究ではオートマトンに基づく並列化はあまり研究を実施しなかった.

(3) m-bridge に基づく木断片化を MapReduce 計算モデル上で行う並列アルゴリズムを設計した. また, MapReduce 計算モデルを実装する Hadoop 上で, Hadoop で導入されている拡張機能も利用して木断片化を効率的に並列計算できることを実際に確認した. ただし本項目については, 今までのところ論文としての発表には至っていない.

(4) XQuery の全体を対象とする並列化を, 研究項目 (1) で得られた成果を拡張する形で実施しようと考えたが, XQuery の並列化における本質的課題と (1) における課題が直接関連しないことが判明し, 本項目については研究を中断した.

(5) 深層強化学習の並列分散実装をサポートする重要なフレームワークである Ray [5] 上で, 生成される学習データを透過的に保存できるようにするライブラリを設計・実装し, 評価を行った. 本成果は, 情報処理学会研究会などで発表した.

#### < 引用文献 >

- [1] R. Bordawekar, et al.: To Parallelize or Not to Parallelize: XPath Queries on multi-core Systems, IBM Research Report RC24926 (W0912-136), 2009.
- [2] V. Braganholo and M. Mattoso: A Survey on XML Fragmentation. ACM SIGMOD Record, 43(3):24-35, 2014.
- [3] W. Hao and K. Matsuzaki: A Partial-Tree-Based Approach for XPath Query on Large XML Trees. Journal of Information Processing, Vol. 24, No. 2, pp. 425-438, 2016.
- [4] 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, No. 3, pp. 51-62, 2007.
- [5] P. Moritz, et al.: Ray: A Distributed Framework for Emerging AI Applications. 13th USENIX Symposium on Operating Systems Design and Implementation, 2018.

5. 主な発表論文等

〔雑誌論文〕 計2件（うち査読付論文 1件/うち国際共著 0件/うちオープンアクセス 2件）

1. 著者名 IWASAKI HIDEYA, EMOTO KENTO, MORIHATA AKIMASA, MATSUZAKI KIMINORI, HU ZHENJIANG	4. 巻 32
2. 論文標題 Fregel: a functional domain-specific language for vertex-centric large-scale graph processing	5. 発行年 2022年
3. 雑誌名 Journal of Functional Programming	6. 最初と最後の頁 -
掲載論文のDOI（デジタルオブジェクト識別子） 10.1017/S0956796821000277	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 -

1. 著者名 Shigeyuki Sato, Wei Hao, Kiminori Matsuzaki	4. 巻 abs/1806.07728
2. 論文標題 Parallelization of XPath Queries Using Modern XQuery Processors	5. 発行年 2018年
3. 雑誌名 arXiv	6. 最初と最後の頁 -
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 無
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 -

〔学会発表〕 計6件（うち招待講演 0件/うち国際学会 2件）

1. 発表者名 飯谷拓晃, 松崎公紀
2. 発表標題 分散強化学習におけるデータ保存ライブラリの実装
3. 学会等名 情報処理学会第158回システムソフトウェアとオペレーティング・システム研究発表会
4. 発表年 2023年

1. 発表者名 飯谷拓晃, 松崎公紀
2. 発表標題 分散強化学習におけるデータ保存ライブラリの実装
3. 学会等名 令和4年度電気・電子・情報関係学会四国支部連合大会
4. 発表年 2022年

1. 発表者名 飯谷 拓晃, 松崎 公紀
2. 発表標題 クエリ分割並列化による並列XPathクエリのXMLデータベースBaseXにおける評価
3. 学会等名 情報処理学会第83回全国大会
4. 発表年 2021年

1. 発表者名 Wei Hao, Kiminori Matsuzaki, Shigeyuki Sato
2. 発表標題 A Dual-Index Based Representation for Processing XPath Queries on Very Large XML Documents
3. 学会等名 10th EAI International Conference, CloudComp 2020 (国際学会)
4. 発表年 2020年

1. 発表者名 Shigeyuki Sato, Wei Hao, Kiminori Matsuzaki
2. 発表標題 Parallelization of XPath Queries Using Modern XQuery Processors
3. 学会等名 European Conference on Advances in Databases and Information Systems (ADBIS 2018) (国際学会)
4. 発表年 2018年

1. 発表者名 Fangyuan Liao, 松崎 公紀
2. 発表標題 Experimental Analysis of Prefix-sums Implementations on Multicore Clusters
3. 学会等名 The 1st cross-disciplinary Workshop on Computing Systems, Infrastructures, and Programming (xSIG 2017), Poster
4. 発表年 2017年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

-

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
--	---------------------------	-----------------------	----

7. 科研費を使用して開催した国際研究集会

〔国際研究集会〕 計0件

8. 本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関
---------	---------