2版

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 １２ 日現在

機関番号：６２６１５
研究種目：挑戦的萌芽研究
研究期間：2015～2017
課題番号：１５Ｋ１２０１１
研究課題名（和文）Structural Recursion on Bulk Synchronous Parallelism for Efficient Large-Graph Querying

研究課題名（英文）Structural Recursion on Bulk Synchronous Parallelism for Efficient Large-Graph Querying

研究代表者
胡　振江（HU, Zhenjiang）

国立情報学研究所・アーキテクチャ科学研究系・教授

研究者番号：５０２９２７６９

交付決定額（研究期間全体）：（直接経費）　　2,000,000円

研究成果の概要（和文）：大規模グラフを効率的に処理するために高水準の並列プログラミングモデルの開発が必要である。本研究では、構造的再帰関数の高水準の表現力とバルク同期並列計算モデルに基づく実用的なシステムPregelを融合することで、高水準で効率的な大規模グラフ問合せを実現する新しい手法を提案した。具体的には、分解可能な構造的再帰関数をPregelに効率的にマップすることを示し、それに基づいた宣言的グラフ問合せ言語UnQLの並列化に成功した。また、リモート読取りと書込みをサポートするvertex-centricグラフ計算モデルを提案し、ドメイン固有言語Palgelを設計しPregelで実現し、その有効性を示した。

研究成果の概要（英文）：The ever-increasing size of graph data today creates a critical need for programming models that can deal with large graph efficiently. In this research, we proposed a novel solution to efficient large-graph querying, by combining the expressive power of structural recursion with Pregel, a popular system based on Bulk Synchronous Parallelism for large scale graph processing. We showed that any decomposible structural recursion can be efficiently mapped to Pregel, based on which, we succeeded in parallelization of UnQL, a declarative graph query language. Moreover, we designed and implemented a new domain specific language, called Palgel, which can support both remote reads and writes, allow programmers to use a more declarative syntax called chain access to directly read data on remote vertices, and be efficiently implemented over Pregel.

研究分野：計算機科学、プログラミング言語、関数プログラミング、並列プログラミング、双方向変換

キーワード：大規模グラフ　構造的再帰　グラフ問い合わせ　並列プログラミング

１．研究開始当初の背景

Graph structure can scale naturally to large datasets, as it does not require expensive join operations. A complex query on a large graph is often very expensive in computation. Parallelism is the main technique to increase computation performance.

The Bulk-Synchronous Parallel (BSP) model introduced by Leslie Valiant simplifies significantly design of parallel algorithm by using synchronization barriers. Based on BSP, Alexander Tiskin proposed several graph algorithms for the fundamental problems like algebraic paths in acyclic graphs, algebraic paths in acyclic graphs, algebraic paths in acyclic graphs and minimum spanning tree computation. BSP has also been implemented in functional language and been extended to hierarchical environment. Inspired by BSP, Google introduced Pregel that provides the possibility to implement efficient algorithms to solve more complex large-scale graph problems, even the optimal algorithms still need to be studied case by case.
However, it is known to be difficult to program directly on Pregel-like framework, and there is no effective solution to this problem.

２．研究の目的

This research aims to propose a novel solution to efficient large-graph querying, by combining the expressive power of structural recursion with Pregel, a popular system based on Bulk Synchronous Parallelism for large scale graph processing. It will make graph querying be declaratively specified, systematically optimized, and efficiently computed with Pregel. We expect that success of this work would contribute to the future goal of a structural methodology for large-graph parallel programming.

３．研究の方法

The project aims to develop a method and a novel software framework for efficient large-graph querying under a distributed system. It is split in the following four steps.

(1) Prototyping and validation a poof of concept of evaluation of bulk semantics over bulk synchronous parallelism.

(2) Improvement and implementation of the poof of concept on large-scale graph processing system GPS, build a parallel query engine.

(3) Improvement of the engine to support very large graph.

(4) Extension of the engine to support variants of structural recursion query. Each step includes theoretical study, coding and experimentation in order to validate the concept. Therefore each milestone can be checked to avoid big error.

４．研究成果

Our main results consist of three parts.
First, by focusing on analysis of structural recursion to find a class of structural recursions that can bridge between high-level graph queries and low-level Pregel implementation, we show that any decomposable structural recursion can be efficiently mapped to Pregel. As a result, we designed and implemented a graph transformation framework on top of Pregel. Our framework accepts graph transformations written in a domain-specific language for describing decomposable structural recursions as its input and automatically compiles these transformations into an efficient Pregel implementation.

Second, we extended the above solution to a framework that can take high-level graph queries in UnQL as input for relaxing complexity of designing structural recursive functions. The gap between large graph processing platform and high-level declarative querying language is thus filled by our solution. A high-level graph query written by an end-user is transformed systematically into our internal algebra with a set of structural recursive functions in UnCAL, then a Pregel program will be generated by using our parameterized Pregel algorithms to guarantee the efficiency of the querying evaluation. More specifically, our results are as follows.

(1) We identified monadic queries, a useful subclass of UnQL queries that can be translated into parallel-efficient structural recursive functions.

(2) We proposed an approach, using pattern trees to describe the relationship between graph variables of

queries, to translating all monadic queries into structural recursive functions in a systematic way.

(3) We used real big datasets to validate our graph querying framework. Both correctness and scalability were experimented, and the experimental results show that our solution may outperform an existing industrial solution for complex queries.

Third, we extended our system further with a more powerful domain specific language that supports remote data access for the vertex-centric graph processing. More concretely, our results are as follows.

(1) We proposed a new high-level model for vertex-centric computation, where the concept of algorithmic supersteps is introduced as the basic computation unit for constructing vertex-centric computation in such a way that remote reads and writes are ordered in a safe way.

(2) Based on the new model, we designed and implemented Palgol, a more declarative and powerful DSL, which supports both remote reads and writes, and allows programmers to use a more declarative syntax called chain access to directly read data on remote vertices, with a new logic system to compile chain access to efficient message passing where the number of supersteps is reduced whenever possible.

(3) We demonstrated the power of Palgol by working on a set of representative examples, including the Shiloach-Vishkin connected component algorithm and the list ranking algorithm, which use communication over dynamic data structures to achieve fast convergence.

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 3 件）

(1) Le Duc Tung, Zhenjiang Hu, Towards Systematic Parallelization of Graph Transformations over Pregel, International Journal of Parallel Programming (IJPP), 査読有, Volume 45, Issue 2, 2017, pp.320-339, https://doi.org/10.1007/s10766-016-0418-5

(2) Chong Li, Le Duc Tung, Xiaodong Meng, Zhenjiang Hu, Let High-level Graph Queries Be Parallel Efficient: An Approach Over Structural Recursion On Pregel, IPSJ Transactions on Programming, 査読有, Vol.24, No.6, 2016, pp.928-936, http://doi.org/10.2197/ipsjjip.24.928

(3) Zhenjang Hu, John Hughes, Meng Wang, How Functional Programming Mattered, National Science Review, Oxford Journal, 査読有, Volume 2, Issue 3, 2015, pp.349-370, 10.1093/nsr/nwv042

〔学会発表〕（計 6 件）

(1) Yongzhe Zhang, Hsiang-Shang Ko, Zhenjiang Hu, Palgol: A High-Level DSL for Vertex-Centric Graph Processing with Remote Data Access, 15th Asian Symposium on Programming Languages and Systems (APLAS 2017), 2017

(2) Kento Emoto, Kiminori Matsuzaki, Zhenjiang Hu, Akimasa Morihata, Hideya Iwasaki, Think Like a Vertex, Behave Like a Function!―A Functional DSL for Vertex-centric Big Graph Processing--, 21st ACM SIGPLAN International Conference on Functional Programming (ICFP 2016), 2016

(3) Chong Li, Le Duc Tung, Xiaodong Meng, Zhenjiang Hu, Derivation of Parallel-Efficient Structural Recursive Functions from Declarative Graph Queries (Poster), 31st ACM/SIGAPP Symposium on Applied Computing (SAC 2016), 2016

(4) Le Duc Tung, Zhenjiang Hu, Towards Systematic Parallelization of Graph Transformations over Pregel, 8th International Symposium on High-level Parallel Programming and Applications (HLPP 2015), 2015

(5) Le Duc Tung, Pregel meets UnCAL: a Systematic Framework for Transforming Big Graphs, 31st International Conference on Data Engineering Workshop (ICDEW 2015), 2015

(6) Chong Li, Le Duc Tung, Xiaodong Meng,

Zhenjiang Hu, Processing UnQL Graph Queries with Pregel, 日本ソフトウェア科学会第32回大会, 2015

６．研究組織

(1)研究代表者

胡　振江（HU, Zhenjiang）
国立情報学研究所・
アーキテクチャ科学研究系・教授
研究者番号：50292769

(2)研究分担者

LI, Chong
国立情報学研究所・
アーキテクチャ科学研究系・特任研究員
研究者番号：50745312