要 旨

並列分散処理向け大規模 XML 木の分割に関する研究

川村 高之

近年，データの大規模化に伴い XML などの木構造のデータも大規模化している．大規模化したデータを効率良く処理する手法の一つとして分散並列処理があるが，その最初のステップとしてデータの分割が必要である．XML などの木構造を用いる処理はノード間の親兄弟関係に依存している場合があるため，分割時にノード同士の関係に気を配らなければならない．そのような性質を持つ木の分割法として，m-bridge による分割が知られている．任意の形の木に対して，m-bridge は部分木に対する単純な計算で求めることができる．しかし，一般的な浅い XML 文書に対して m-bridge 分割を行うと，細かすぎる分割となる場合があるなど，適切な分割にならないことがある．

本研究では，この問題を解決するため XML 木に 1 対 1 対応する二分木に対して m-bridge を適用することで XML 木を分割するアルゴリズムを示す．実際に，SAX Parser ライブラリを用いて提案アルゴリズムを実装する．SAX による XML 読み出しの手順にあわせて二分木変換と分割を同時に行うため，二分木の変換方法を工夫している．そのプログラムを用いて実験を行い XML 形式のデータの二分木に基づく分割の性質を調査する．また，既存の m-bridge による分割と提案手法の二分木表現上の m-bridge による分割で得た分割データに対してそれぞれ Hadoop を用いて並列処理を行い処理時間を比較した．さらに，分割データに対するクエリの適用といった，分割した木データの利用方法について議論する．

キーワード　XML，分散処理，データ分割，二分木表現，SAX，Hadoop

Abstract

Study on Dividing Huge XML Trees for Parallel and Distributed Computing

Takayuki Kawamura

Recently, tree data such as XML trees are getting larger and larger. Parallel and distributed processing is a promising way to deal with those big data, but we need to divide the data at the first step. Since computation over trees often requires relationship between parent and children and/or among siblings, we should take care of such relationship. There is a technique called "m-bridge" for dividing trees. We can easily compute m-bridges for trees of any shape. However, the division by the m-bridge technique is sometimes unsatisfactory for shallow XML trees.

In this study, we propose a tree division method for XML trees in which we apply the m-bridge technique to a one-to-one corresponding binary tree. We implement the tree division algorithm using the SAX Parser. An important point in our algorithm is that we transform and divide XML trees in the order that the SAX parser reads the trees. We make experiments and discuss the properties of the proposed tree division algorithm. We also make experiments about the performance on Hadoop for the data divided with two algorithms. In addition, we discuss how we can use the divided trees with query examples.

key words XML, distributed computing, data division, binary-tree representation, SAX, Hadoop