3Q-1

データベースプロセッサＲＩＮＤＡの大容量ソート処理方式

佐藤 哲司、武田 英昭、中村 敏夫、速水 治夫
ＮＴＴ情報通信処理研究所

1．はじめに

データベースプロセッサＲＩＮＤＡの１つの構成要素であるＲＯＰ(関係演算機構)は、専用ハードウェアを用いてソートや結合演算等を高速化している[1,2]。

本稿では、キー長やキー数に柔軟に対処できる大容量ソート処理方式について述べる。

2．ソータ構成法

ＲＯＰは、図１に示すようにチャネルを介して本体装置に接続され、ホストから送られてくる表の各行からキーを抽出してソートする機能を有する。10万件を越えるキーを小型な装置で高速にソートするために、マルチウェイマージ処理を多段階に繰り返す多段マージソート法[3]を採用している。

2-1. 基本動作

図２はＲＯＰソート部の構成であり、比較転送ユニットを１次元アレイ状に配置したソートアレイを用いて、最大ｋ件のキーを並列に比較できる。以下に、本ソート回路を用いたソート処理の手順を示す。

①初期ソート段階：各行から抽出したキーを、ソートアレイを用いてｋ件づつ連続的にソートし、結果を大きさｋのソート列としてＲＯＰメモリに格納する。キー抽出→ふるい落とし→ソート→行およびソート列の格納の各処理をパイプライン的に実行し、ホストからの表(行)の転送時間で初期ソート段階の処理を完了する。

②中間マージ段階：ＲＯＰメモリに格納されたソート列を最大ｋ本づつマージして、結果をＲＯＰメモリに格納する処理を、全ソート列の本数がｋ以下になるまで繰り返す。 N件のキーをソートするのに要する中間マージ段数mは $m=\lceil \log_k N \rceil - 2$ である。ここで$\lceil x \rceil$は、ｘを超える最小の整数を示す。なお、m≦0の場合は中間マージ処理は行わずに、出力マージ段階に移行する。

③出力マージ段階：ＲＯＰメモリに格納されたソート列をマージし、キーの並び順に対応して行データを出力する。ソートした表は、ページ単位に編集して、本体装置に返送する。

本ソート法の特徴は、ＲＯＰメモリを繰り返し使用して大量のキーをソートできること、ソートアレイでｋ個のキーを並列に比較するため、一定のマージ処理速度を保ったままで、マージウェイ数ｋを大きくできることである。

ＲＯＰでは、チャネル転送時間に重畳して、初期ソート段階と出力マージ段階の処理を行う。このため、本体装置からみてソートに必要な時間は、中間マージ段階の処理時間となる。この時間は、中間マージ段数に依存し、比較転送ユニット数を増やすことによって削減できる。

2-2. ユニット連結動作

データベースプロセッサは、整数(二進数)のような短いキーから、文字列等の比較的長いキーまで、幅広い範囲の長さのキーをソートできなければならない。ＲＯＰのソート部を構成するソートアレイは、１次元配置した比較転送ユニットの各メモリにキーを格納して比較を行うことから、各ユニットのメモリ容量でソートできるキー長の上限が規定される問題がある。

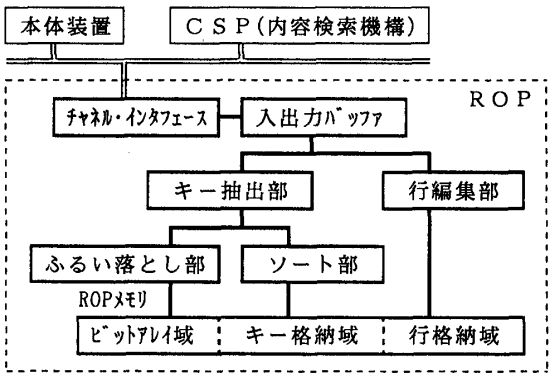

図１ ＲＯＰの構成

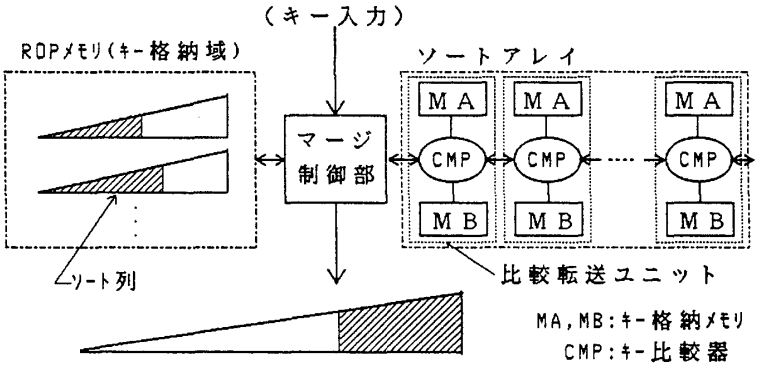

図２ ソート部の論理構成

Mass-storage Sorting Methods in RINDA : Relational Database Processor
Tetsuji SATOH, Hideaki TAKEDA, Toshio NAKAMURA and Haruo HAYAMI
NTT Communications and Information Processing Laboratories

キー長の上限を拡張する方式として以下の２案がある。

①キーの最大長に合わせて、各比較転送ユニットのメモリ容量を大きくする方式。

②各ユニットのメモリ容量は一定として、キー長に応じて複数ユニットを連結して動作させる方式。この方式では、キーは複数の比較転送ユニットに分割格納される。

ＲＯＰでは以下の理由により、案②を採用した。

ⓐ総ハード量を一定とすると、キー長が短い範囲で案②の方が、より多くのユニットを実現できる。一般に、キー長が短い表に対するソートの使用頻度が高いと考えられるため、キー長が短い領域でより高い性能が実現できる案②が適する。

ⓑキーの最大長は、日本語文字列場合でも１００文字程度と考えられる。キー長の上限を２５０バイト程度とした場合、数個のユニットを連結することで実現できる。本ソートアレイでは、数個のユニットの連結動作は容易であり、制御回路も比較的簡単に実現できる。

ⓒユニット連結を行うと、実効の比較転送ユニット数が少なくなるためマージウェイ数が減少する。しかし、キー長(行の長さ)が長くなれば、ＲＯＰメモリに格納できるキー数が相対的に減少する。このため、通常２段の中間マージで、ＲＯＰメモリに格納できる件数のキーをソートできる。

３．大容量ソート処理方式

多段マージを採用したＲＯＰソータでは、基本的にソート件数に上限はないが、実装されているＲＯＰメモリ容量によりソート容量が定まる。ＲＯＰのソート容量を超える表のソートは、ＲＯＰがオーバフローするまでの範囲で分割ソートした後に本体装置においてマージする。ここでは、ＲＯＰにより分割ソートを行う場合に必要となる機能について述べる。

３-1．オーバフロー検出法

キー格納用と行格納用とに共用しているＲＯＰメモリは、可変長の長さを持つ行を扱うために、両者の境が動的に変化できる構成としている。この結果、ＲＯＰメモリが真に満杯になるまでソートできる件数を増大できるが、ソータのオーバフローは行の長さに依存し、事前にソート可能件数を知ることはできない。更に、ソート前処理としてふるい落とし処理を行う場合もあることから〔2〕、ソート処理過程で常にオーバフローの検出を行う必要がある。

キー格納域には、多段マージ処理を行うためのマージ作業域が含まれるが、その大きさはキーの件数から算出できる〔3〕。一方、行格納域の大きさは、格納する行毎にその長さが可変であるため、実際に行を格納するまでその長さを知ることはできない。このため、所定の件数のキー入力ごとに、ソート作業域を含めたキー格納域の大きさを算出し、行を格納する毎にオーバフローの判定を行なう。

３-2．ＲＯＰ入力の再開法

二つの表の結合は、ソートマージ結合法を基本として処理し、個々の表をＲＯＰでソートする段階で、ビットアレイを用いて結合する可能性のない行をふるい落として、結合処理の演算量を削減している〔2〕。

オーバフローが発生しない通常の使用形態では、ＲＯＰの入出力処理は、表を単位として完結している。この特徴を利用して、通常、ＲＯＰからソート結果を出力する処理の裏でビットアレイを初期化して、初期化に要する時間をゼロにしている。

ビットアレイの設定、参照およびソート処理は、ＲＯＰ入力操作と並行して処理することから、オーバフローが発生した場合に、ビットアレイに対する処理が継続できるようにＲＯＰ入力を再開しなければならない。

ビットアレイを操作中にオーバフローが発生するのは、ソート処理と組み合わせて、ビットアレイの設定あるいは参照処理を行う場合である。いずれの場合も、ビットアレイの状態を保持したままで、一旦、ソート結果を出力してＲＯＰメモリのキー格納域と行格納域を空にしてから、ＲＯＰ入力を再開する。

このため、ＲＯＰでは、ビットアレイの設定・参照処理を行っている際にオーバフローが発生した場合、ビットアレイのクリア契機を最後の参照ソートの出力時まで遅らせることで、オーバフロー時であっても処理を継続できる方式としている。

４．まとめ

ＲＯＰの主要な構成要素であるソート部の構成法に関して、キー長やキー数に柔軟に対処できる大容量ソート処理方式について述べた。ソートアレイを構成するユニットを連結することで、ソートできるキーの長さを約２５０バイトまで拡張した。ＲＯＰメモリの行格納域とキー格納域の境界を動的に変化させ、１０万件以上の行からなる表を一度にソートできる方式とした。更に、ＲＯＰがオーバフローするような大規模データベースをソートする場合であっても、ビットアレイの設定・参照処理を継続できる構成として、ＲＯＰを繰り返し使用して分割ソートが行える方式とした。

〔参考文献〕

〔1〕速水、井上他：ﾘﾚｰｼｮﾅﾙﾃﾞｰﾀﾍﾞｰｽﾌﾟﾛｾｯｻＲＩＮＤＡの ｱｰｷﾃｸﾁｬ、 情処研報 88-CA-73-12、1988

〔2〕武田、佐藤他：データベースプロセッサＲＩＮＤＡの関係演算方式、情処37全大5Q-6、1988

〔3〕佐藤、武田他：大容量データベース処理に適したソート手法、 信学技報 DE88-1、1988