

3F-1

関係代数による問合せ最適化に関する
 一考察—並列処理に於ける最適化—
 鈴木重信・喜連川優・田中英彦・元岡達
 (東京大学工学部)

1.はじめに-----近年研究が盛んであるデータベースマシンにおいても従来のソフトウェアによるDBMS同様、問合せ最適化は重要な問題である。問合せ最適化は、i)オペレーションの高速化, ii)オペレータ結合レベルでの最適化, iii)冗長性除去の3レベルに分けられる。データベースマシンでは、i)はハードウェアで実行しているので増進化されていると考えられ、ii)のみを考察するだけで十分である。又、処理が高速化されてもオペレータ結合レベルでの最適化の効果が導かれるわけではない。そこで本研究では、並列処理可能な環境として二種類のデータベースマシンのモデルを設定し、それに与える関係代数のオペレータ結合レベルでの最適化(特に2項演算子について)を考えた。従来あまり触れられていないunionやdifferenceについても、これを含めて考えた。

2.マシン・モデル-----我々はこれまでデータベースマシンGRACE^{[1][2]}の開発を行ってきたが、ここでは $O(n)$ モデル(二項演算子の処理に $O(n)$ の処理時間がかかるマシンのモデル)と $O(m)$ モデル(処理時間 $O(m)$ のモデル; m はメモリモジュール[M.M.]の容量)の2つのモデルを考える。GRACEは、ディスクモジュール, M.M., プロセッシングモジュール[P.M.]及びそれらを結ぶリングバスから成っている。 $O(n)$ モデルでは、P.M.は R_1 の比較レジスタを内蔵した連想プロセッサであり、リレーション R_1, R_2 に対して「 $R_1 \text{ op } R_2$ 」を行なうのに「 $C/N_1 M_2 m / R_1 P$ 」(C :比例定数, N_1, R_1 のタプル数, $M_2: R_2$ のページ数, P : P.M.台数)だけの時間がかかる。 $O(n)$ モデルではP.M.は、一定量のデータを入力データ流に同期してソートするソートモジュール[S.M.]を内蔵している。この時は、M.M.中のハッシュユニットでリレーションをP.M.の容量に見合うようバケット化することで、オペランドリレーションがM.M.上にある限り、すべてのオペレーションを(リレーションサイズに係わりなく)M.M.の容量に比例した時間で終了できる。

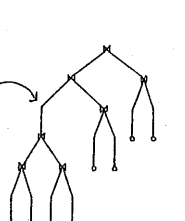
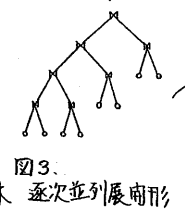
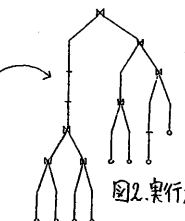
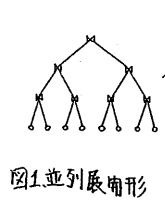
3.最適展開の要因----- $O(n)$ モデルでは、リレーションが十分に大きければ、同時には1つのオペレーションのみ実行した方が全体の効率が良くなるため、総比較回数を減少させるようにオペレータの結合を考えなければならぬ。 $O(m)$ モデルでは、すべてのオペレーションがリレーションサイズに係わりなく一定時間で終了するため、その一定時間を1phaseと見て、全体の問合せを実行するのに要するphase数を最小にすればよい。この時できるだけ多くのオペレーションを並列に実行して最小化する。Stagingも1phaseで終了すると考える。ある展開形の処理に必要なphase数を見るには、図2, 3のような実行木の高さを見ればよい。M.M.台数が無限にあるとすれば、高さ最小のtree(図1)に展開すべきであったが、実際はM.M.が有限なため、図3のような逐次並列展開形(実行木にそのまま対応するもの)にする必要がある。 $O(m)$ モデルにおける問合せ最適化のためには、i)中間リレーション総量の最小化(生成されたリレーションはM.M.を占めるため), ii)処理待ち時間(M.M.に存在するリレーションの相手がないため待ち時間)の最小化, iii)2ヶ境境界性の考慮(同一レベルにあるリレーション数を2のべき乗に近づける), が必要である。

4.最適化の必要性-----二項演算子について表1の変換則を考へ、すべての等価な表現の処理コストを比較することにより最適展開形を求めてみた。結果リレーションサイズの予測はアトリビュートの最小値, 最大値を使って行なった。リレーション数と以下の

様々な問合せの6例に対して表2のデータで最適展開形を求め、最適化を行なわな
い時とのコストの比を求めると表3のようになった。(但し、 $O(m)$ モデルで最適化を
行なわないと処理不能になった割合が不能率)この結果次のことがわかった。i)一般
に分配則に関し結合した(表1の左辺)方が分配する(右辺)よりも有利である。ii) $O(m^2)$
モデルでのjoinの順序の選が方は効果が大きい。(INGRESのdecomposition[3]) iii) $O(m)$
モデルで単に並列展開を考えただけでは不十分である。iv) $O(m)$ モデルでは最適化
の効果がかなりある。v) $O(m)$ モデルでは最適化が実行不能展開形除去に役立っている。
5. 近似最適展開生成アルゴリズム-----リレーションが多くなると、全写価表現の比較
は奥用的でない。そのため、与えられた問合せに対し多項式関数の比較回数で終了
するアルゴリズムを考えた。基本方針は次の通りである。i)分配則の適用で済んだ個
所を l とすると、あるheuristicsに従い、必要と思われた $O(l)$ 個の異なる展開形を生
成し、処理コストの比較をする。ii)join(union)の系列については、他の部分に独
立に順序を決定する。iii) $O(m)$ モデルではstagingをシミュレートしながらiii)を行なう。特に
重要なjoinの順序については、 $O(m^2)$ モデルでは $|R_i|/(1-|R_i|/|R_j|)$ の最も小さいも
のから結合していき、 $O(m)$ モデルではquery graph上で結果リレーションサイズの大きくなった
edgeを切りながら2つずつのmodeの組を作り、それを結合する。上記の6例に対し
このアルゴリズムを適用すると、完全な最適展開形を生成しなかったものは $O(m^2)$ モデ
ルでは3例のみ(但し、そのコストの差は数%以下)、 $O(m)$ モデルではゼロであった。

6. おわりに-----データベースマシンの環境下でのオペレータ結合レベル最適化に
ついて検討したが、今後、データベースマシンの研究の発展とともに、オペレーション
の高速化だけでなく、各マシンモデルに対して問合せ最適化の研究が必要となっ
てくるだろう。今後、i)実際に使われた問合せをベースにした解析、ii)単項演算
子を含めた解析、iii)マルチユーザの環境における最適化、iv)結果リレーションのよ
りよい予測(処理に応じた動的な変更も含めて)等に関する考察が必要となる。

参考文献 ①喜連川他, "可変構造多重処理データベースマシンの構成", 信学技報, EC80-51, 1980
②喜連川, 鈴木他, "HashとSortによる関係代数マシン", 信学技報, EC81-35, 1981
③E. Wong, K. Youssefi, "Decomposition - a Strategy for Query Processing", ACM TODS, vol 1, no 3, 1976
④P.G. Selinger et al, "Access path Selection in a Relational Database Management Systems", Proc. ACM
SIGMOD, 1979



$R_1 \bowtie R_2 = R_2 \bowtie R_1$
$(R_1 \bowtie R_2) \bowtie R_3 = R_1 \bowtie (R_2 \bowtie R_3)$
$R_1 \cup R_2 = R_2 \cup R_1$
$(R_1 \cup R_2) \cup R_3 = R_1 \cup (R_2 \cup R_3)$
$(R_1 \cup R_2) \bowtie R_3 = (R_1 \bowtie R_3) \cup (R_2 \bowtie R_3)$
$(R_1 - R_2) \bowtie R_3 = (R_1 \bowtie R_3) - (R_2 \bowtie R_3)$
$(R_1 \cup R_2) - R_3 = (R_1 - R_3) \cup (R_2 - R_3)$
$R_1 - (R_2 \cup R_3) = (R_1 - R_2) - R_3$

表1

タプル数: 平均/8/8の指数分布
アトリビュトの最大値: 平均2432の指数分布
アトリビュトの最小値: 平均10の指数分布
タプル長: 一定
メモリモジュール容量: 100タプル
メモリモジュール台数: 100台

表2

joinのみから成る問合せ	
最適/lambda順逐次展開($O(m^2)$)	0.41
最適/lambda順逐次展開($O(m)$)	0.78
不能率	8%
最適/lambda順並列展開($O(m)$)	0.9
不能率	15%
分配則のかさむ問合せ	
最適/lambda順分配型($O(m^2)$)	0.68
最適/lambda順結合型($O(m^2)$)	0.76
最適/lambda順分配型($O(m)$)	0.81
不能率	12%
最適/lambda順結合型($O(m)$)	0.92
不能率	24%

表3