5H-4 データストリーム処理に基づく逐次型ＧＲＡＣＥの性能評価

中山雅哉，喜連川優，高木幹雄
東京大学生産技術研究所

1．はじめに

現在我々は，データストリームを処理の単位として取り扱う関係データベースシステムの研究・開発を進めている。本システムでは，従来から研究を進めてきた関係データベースマシンＧＲＡＣＥの構成と制御方法を汎用計算機上に有効に取り入れた実装が成されており，負荷の重い結合演算に対しても，動的クラスタリングと，線形時間でのソート処理により，演算の対象となるリレーションの大きさに比例する時間で応答することが可能となる。本稿では，動的クラスタリングを中心として汎用計算機上に実装した試作システムの結合演算処理の性能について報告する。

2．試作システムの構成と処理手順

MELCOM80/500 (OS:DPS10) 上に実装した試作システムは，図１に示す様に，複数のプロセスより構成されている［１］。各プロセスの働きは，ＧＲＡＣＥの各モジュールとほぼ同等であり，システム全体は，Ｃ－ Processによる逐次型ＧＲＡＣＥとしての制御により動作する。すなわち本システムでは，処理の対象となるリレーションが，データストリームとしてディスクから読み込まれ，各プロセス間を通信する間に必要な演算が施されることになる。

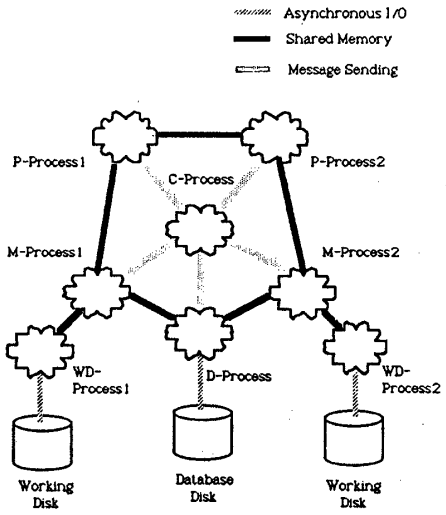

図１　試作システムの全体構成

例えば，図２(a) に示す問合せに対しては，以下の３つのタスクに処理が分割されて実行される（図２(b)）。

①リレーションＳをディスクから読み込むと共に，Ｄ－Process 内で S.attr1 < CONSTなる選択演算を施し，Ｍ－ Process1 に転送する。Ｍ－ Process1 では，結合属性(S.attr1) をキーとしてハッシュ操作を施し，複数のバケット空間に分割して管理を行う。

②リレーションＰに関しても同様にして，P.attr1 に関する選択処理・ハッシュ処理を施し，①と同一のバケット空間に格納する。

③各バケット毎に，Ｐ－ Process1 でソート処理を，Ｐ－ Process2 で結合処理を施して，Ｍ－ Process2 のバケット空間に結果を格納する。

ここで，Ｃ－ Processによる制御は，各タスク単位に１度ずつ行われるだけであり，ＤＩＲＥＣＴ［２］にみられた様にページ単位に制御の介入が起ることはない。従って，制御オーバーヘッドがシステムの性能を制限することはなくなる。

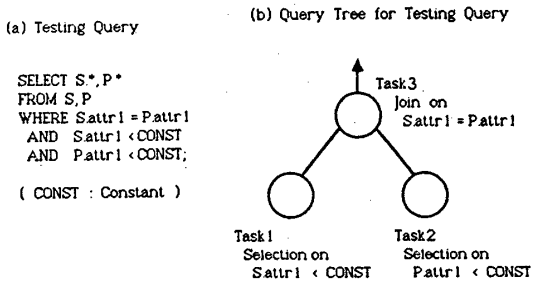

図２　問合せ式の構成と処理手順

3．性能評価モデル

試作システムの性能を評価するのに，図３に示す様な５属性から成るタプル長64バイト，タプル数10,000のリレーションを使用した。

また，性能評価には図２の問合せを用いた。その実行手順は，先に記した通りである。基本的な結合演算を評価の対象としたのは，この演算の処理負荷は非常に重く，現在の商用システムにおけるシステムの性能を大きく左右する演算であるためで，この演算の解析，評価を行うことは，より良いシステム構成を考える上で重要なため

Sequential Implementation of GRACE Based on Data Stream Model and its Evaluation
Masaya NAKAYAMA, Masaru KITSUREGAWA, Mikio TAKAGI
Institute of Industrial Science, University of Tokyo

である。

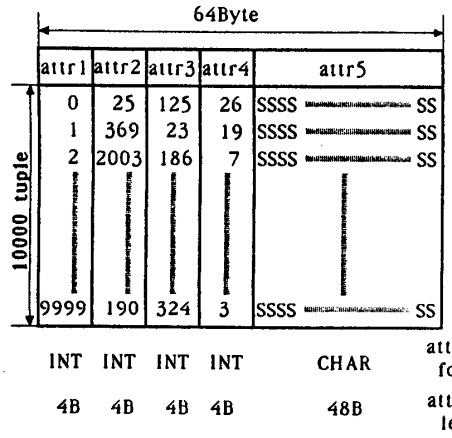

図３　評価に用いたリレーションの構造

４．評価結果

試作システムについて，図２(a) の CONSTを０～10,000まで変化させた時の，応答時間，Ｐ－ Process1 のユーザモードＣＰＵ時間を図４に示す。この結果から，Ｐ－ Process1 のユーザモードＣＰＵ時間（ソート処理時間）は，CONST の大きさにほぼ比例した時間ですむが，システムの応答時間は，CONST の大きさ（選択率の大きさ）により性質が異なっており，選択率が小さい時には処理リレーションの大きさに比例しないことが判る。これは，全ＣＰＵ時間に対する各プロセスのＣＰＵ時間を示した表１から明らかな様に，CONST が小さい場合は，ソート処理時間に比べて，Ｄ－ Processでの選択処理時間の割合が大きいためである。

また，表１で注目すべきことは，Ｃ－ Processの処理負荷である。前節で述べた様に，Ｃ－ Processの制御はタスク単位に各プロセスに一度ずつ行われるだけなので，その負荷はシステム全体の負荷に比べて非常に軽いものとなる。従って，ＤＩＲＥＣＴにおいて問題となった，システム制御のためのマシン性能低下は本システムでは起らない。

これらの評価結果から以下の点を改良すれば，システムの性能は更に向上すると考えられる。

(1)現在の試作システムでは，ソフトウェアによりソート処理を行っているが，大規模なリレーションの処理においては，ソート処理の負荷は他の処理に比べ，非常に重いものになっている。これを我々が別に研究を進めているハードウェアソータ［３］を用いて処理する様に改良すれば，システムの性能は，格段に向上することが期待できる。

(2)また現在の試作システムでは，インデクスを用いたリレーション管理を行っていない為に，対象リレーションに対する選択率が小さい問合せに対して，Ｄ－ Processの負荷が比較的高くなっている。これに対して，ＧＫＤ木［４］等を用いて，ディスク内のリレーションをインデクス管理する様に改良すれば，処理対象の小さな問題に対する性能向上も期待できる。

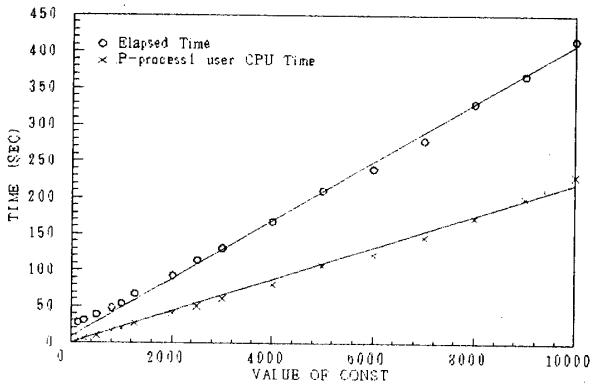

図４　試作システムの結合演算に対する応答時間とＰ－ Process1 のＣＰＵ時間

表１　全ＣＰＵ時間に対する各プロセスのＣＰＵ時間比

CONST 値	Mode	プロセス名称 C	D	M1	WD1	P1	P2	M2	WD2
250	KERN	3.41%	8.70%	1.72%	0.77%	1.70%	0.90%	1.17%	0.71%
	USER	1.79%	50.32%	2.55%	0.32%	19.31%	5.05%	1.29%	0.30%
1250	KERN	1.58%	3.91%	1.44%	1.96%	1.34%	0.58%	0.96%	1.09%
	USER	0.78%	25.12%	4.74%	0.48%	42.71%	10.48%	2.54%	0.31%
10000	KERN	0.23%	0.82%	1.52%	3.35%	1.43%	0.38%	0.91%	1.63%
	USER	0.12%	9.27%	5.72%	0.70%	57.61%	13.02%	2.89%	0.39%

５．おわりに

データストリーム処理に基づいた関係データベースシステム逐次型ＧＲＡＣＥの，試作システムについてその性能評価を行った。現在の試作システムの性能は，それ自身だけでも他の商用システムに比べて数段良い結果が確認されているが，ハードウェアソータを用いた改良により更に大幅な性能の向上が期待できる。現在，ホストマシンとハードウェアソータの結合を進めている。改良を加えた逐次型ＧＲＡＣＥの性能評価については，機会をかえて，報告するつもりである。

参考文献

[1] 中山他，「データストリーム指向関係データベースシステムの構成」，信学技法，EC85-65，1986

[2] H. Boral, et al.「Implementation of the Database Machine DIRECT」，IEEE Transaction on Software Engeneering, 1982

[3] 鈴木他，「超高速（4MB／s）大容量（8MB）ハードウェアソータの実装」情報処理学会第３３回全国大会，5H-3，1986

[4] S. Fushimi, et al.「Algorithm and Performance Evaluation of Adaptive Multidimensional Clustering Technique」，ACM SIGMOD 1985