準接合による ADT 関数を含む問合せの最適化

大保信夫[†] 張暁冬[††]
陳漢雄[†] 藤原譲[†]

抽象データ型 (ADT) の関係データベースへの導入の結果，問合せ最適化に新しい問題点が生じている．計算コストの大きい ADT 関数を含む選択演算は接合演算と同等あるいはそれ以上コストのかかる演算となる．このような状況では選択演算が問合せ処理コストの主要な要素となり，従来の接合演算を主要なコスト要因とみなす最適化技法を適用できない．本論文ではコストの大きい ADT 選択演算を含む問合せに対する新しいアプローチとして，分散環境での最適化技法でよく知られている準接合演算を用いる方法を提案する．本方法は問合せの前段階で準接合演算を用いて，ADT 関数を評価しなければならないタプル数を減らすことにより，全処理コストを減少させるものである．本方法を星型問合せに対して用いた場合のコスト式とそれに基づく最適化手法を述べる．またシミュレーションに基づき本方法が従来の最適化技法に比べ優れた実行プランを生成することを示す．

Using Semijoins to Optimize Queries Including ADT Functions

Nobuo Ohbo,[†] Xiaodong Zhang,[††] Hanxiong Chen[†] and Yuzuru Fujiwara[†]

ADTs (*Abstract Data Types*) have been known as a promising feature for making the relational database meet the requirement of some non-traditional applications, such as CAD/CAM. This extension introduces a new dimension to query optimization. In databases that support ADTs, the execution cost of a selection involving ADT functions (briefly, *ADT selection*) may be computationally expensive. As a result of this, the conventional query optimization method, which considers only joins as the dominant cost factors, can no longer be appropriate for queries involving costly ADT functions. In this paper, we propose an optimization method that takes ADT selections as well as joins into consideration. We show the possibility that, based on the assumption that a costly expensive ADT selection is involved, semijoins, which generally are used in a distributed environment to reduce transmission costs, can be used effectively as a means to reduce the total execution cost in a centralized environment. We discuss the case of the star-query and give an algorithm that generates cost optimal query execution plan. Furthermore, an approximate algorithm is also discussed. Some simulation results show the effectiveness of our optimization method.

1. はじめに

CAD/CAM システムやソフトウェア開発等における関係データベースの利用の高度化に伴い，抽象データ型 (ADT) の導入が有力な手段として認識されてきている[1)~3)]．この ADT の導入は問い合わせ処理，特に最適化に新しい問題を提起している[4),5)]．従来の最適化戦略においては，選択演算，射影演算のコストは接合演算に比べ十分に小さいことが仮定されてきた．この仮定に基づき，選択一射影一接合問合せ中最もコストの高い接合演算に焦点を絞った最適接合順序決定問題に対する研究が行われた[6)~9)]．

しかしながら，計算コストの大きい ADT 関数の評価を含む選択演算のコストが，接合演算のコストを上回ることも起こりうる．これは選択を実行するため各タプルに対して ADT 関数の評価を行わなければならないからである．化合物データベースを例にとると化合物の構造は数学的グラフとして表されるので，ADT "graph" が必要となる．関係Rの属性 CHEM-GRAPH の定義域が "graph" であるとすると選択条件として次のようなものが考えられる．

R. CHEM-GRAPH isomorphic $graph_0$

ここで isomorphic は2つのグラフの同型性を判定する ADT 関数，$graph_0$ は graph 型の定インスタンス（ある与えられたグラフ）である．同型性判定はよ

† 筑波大学電子・情報工学系
Institute of Information Sciences and Electronics, University of Tsukuba
†† 筑波大学工学研究科
Program in Engineering Sciences, University of Tsukuba

く知られているようにきわめて計算コストの大きい関数であり，ノード数の大きなグラフに対する1回の評価は十分ブロックアクセスのコストと比較しうる．したがって上のような選択条件の実行コストは接合演算のコストを上回ることがある．この問題を解決するため二次索引の利用が提案されている[10)~12)]．しかしながら上の例に代表されるような複雑なADTインスタンス，あるいは付随するADT関数，に対する索引を構築することは困難である．その理由としては以下が挙げられる．（1）ADTの定義域に対して線形順序を与える標準的方法がない．（2）長大フィールドにもなる定義域の値に関してその同値性を決定する効率的方法が一般的には存在しない．

Yajimaら[4)]は接合選択率が小さいケースでは，ADT関数を評価しなければならないタプル数を接合演算を用いて減少させることができることに着目し，計算コストの大きい選択の前に接合演算を実行する問合せ実行プラン（QEP）をその探索空間から除去してはならないことを示した．この問題を詳細に調べた結果，分散環境での最適化手法としてよく知られている準接合演算が，より効率的にADT関数の評価回数を減少させうることがわかった．この結果，ADT関数の評価数の減少によるコストの低減が準接合演算に必要とするコストを上回る場合には，問合せの前段階に準接合演算を行うことにより全問合せ処理コストを低減させうる．

本論文では，ADT選択を接合演算とともに主要なコスト要素としてみなす新しい最適化法を提案する．この方法では，選択—射影—接合問合せ処理の前段階で全処理コスト低減に寄与する準接合演算とADT選択演算を実行し，その後接合演算を行う．具体的には実行コストの評価に基づき最適の準接合演算列と接合列を決定する．ここでの最適という用語はコストの最小を意味する．

本方法を中央にADT選択の対象となっている関係（ADT関係），周囲にADT選択には関係しない関係（通常関係）からなる星型問合せ（接合グラフが星型になる問合せ）に関して適用し，最適化アルゴリズムを与える．星型問合せは一般の問合せの解析を行うときに，各ADT選択の影響を局所的に扱うときのベースとなる．このアルゴリズムは通常関係の数をNとすると$O(2^N)$の計算量をもち，Nの数が大きくなると実用性を失う．そこでNの数が大きい場合に対応する$O(N^2)$の計算量をもつ近似アルゴリズムを与える．

われわれはこの方法をADT選択を有するいくつかの星型問合せに適用した．そのシミュレーションの結果は，本方法は従来の問合せ実行プランに比べ優れた実行プランを与えることを示している．とくに，ADT選択のコストが接合演算コストより十分に大きい場合，接合演算の選択率が十分に小さい場合，接合に関与する属性（接合属性）のイメージサイズ（接合属性における異なる属性値の数）が小さい場合は特に優れた実行プランとなる．

本論文の2章では研究の動機となる準接合によるADT関係のタプルの削減に関して簡単な例に基づいて説明する．3章では星型問合せに対する問合せ処理のコスト式を導入し，特にADT関係と，1つの通常関係からなる最も単純な星型問合せに対して準接合演算，ADT選択演算，接合演算の実行順序に関する分析を行う．4章でこのコスト式に基づく最適化アルゴリズムを与える．また関係の数が大きい場合に有効な近似アルゴリズムとその問合せ例に対する適用結果について述べる．最後の5章で結論と今後の課題について述べる．

2. 研究の動機

表1に示す化合物データベース中の2つの関係StructureとCompoundを例にとり，本研究の動機となった準接合演算に基づくADT関係の削減について述べる．Structureは ClassとGraphを属性としてもつ．ここでGraphの定義域はADTであり，isomorphic等のADT関数が付随している．CompoundにはClassとName等の属性がある．このデータベースの典型的な問合せ例が次に示されている．

```
select Compound.Name
from Compound, Structure
where isomorphic (Structure.Graph, graph0)
```

表1 関係例
Table 1 Examples of relation.

(a) Structure

Class	Graph	—
1	g_1	—
1	g_2	—
2	g_3	—
4	g_4	—
2	g_1	—
6	g_2	—
6	g_3	—
4	g_4	—

(b) Compound

Name	Class	—
cn 1	1	—
cn 2	1	—
cn 3	1	—
cn 4	3	—
cn 5	5	—
cn 6	8	—

表 2 接合例と準接合例
Table 2 Examples of join and semijoin result.

(a) Structure ⋈ Compound

Name	Class	Graph	—
cn 1	1	g_1	—
cn 2	1	g_1	—
cn 3	1	g_1	—
cn 1	1	g_2	—
cn 2	1	g_2	—
cn 3	1	g_2	—

(b) Structure ⋉ Compound

Class	Graph	—
1	g_1	—
1	g_2	—

この問合せは，与えられた $graph_0$ と同型なグラフを化学構造としてもつ化合物名を出力するものである．ここで $graph_0$ は graph の定インスタンスを表す．従来の選択を先に実行する実行プラン (QEP) を生成する問合せ最適法では

$$\Pi_{Name}(\sigma_{isomorphic(Graph,graph_0)}Structure) \bowtie Compound$$

という QEP が得られる．$graph_0$ のノード数が大きくなると選択演算の計算コストが主要なコスト要因となり，このコストを下げることが最適化の主要な目標となる．接合選択率が小さいときには Yajima らは以下の QEP が有効であることを示した[4]．

$$\Pi_{Name}(\sigma_{isomorphic(Graph,graph_0)}(Structure \bowtie Compound))$$

これは**表 2** (a)に示すように接合演算の結果，isomorphic を計算しなければならないタプル数が減少したことによる，この QEP が有効であるためには

$$|Structure \bowtie Compound| \leq |Structure|$$

が必要条件となる．ところでこの例でみると接合結果は 6 タプルで Graph インスタンスは 2 個である．一般に ADT インスタンスに対して同値性を調べる 標準的操作は存在しないので 6 タプルに対して isomorphic の評価を行わなければならない．Graph インスタンスが重複して現れる理由は Compound の Class 属性値に重複があることによる．$Structure \ltimes Compound$ を実行した結果は表 2 (b)に示されているように 2 タプルとなる．このことは

$$\Pi_{Name}(\sigma_{isomorphic(Graph,graph_0)}(Structure \ltimes Compound)) \bowtie Compound$$

という QEP では接合演算を先行したときの ADT 関数の評価回数が 6 回であったものが 2 回にまで減少していることがわかる．ADT 選択演算のコストは選択演算を適用するタプル数に比例することは明らかなので，準接合による ADT 選択演算のコストの削減は，準接合に必要なコストを考慮しても全問合せ処理コストの削減をもたらすことが可能となる．特に接合属性の値の重複度が大きいときには，接合選択率に対する Yajima らの必要条件が成立しない場合にも準接合演算により全コストの削減が可能となる．

3. コスト・モデル

この章では準接合，ADT 選択，接合に対するコスト式を定め，ADT 関係を中心ノードとし，通常関係を葉ノードとする星型問合せに対するコストモデルを導く．

3.1 記　　法

星型問合せ (**図 1**) の中心ノードを R_0，葉ノードを $R_1, \cdots, R_n$ とする．このとき R_0 は ADT 関係，R_i $(1 \leq i \leq n)$ は通常関係とする．R_0 の属性 B の定義域が ADT であり，その ADT 上で関数 f が定義されているとする．各関係 R_i の R_0 との接合に関与する属性をおのおの A_i とする．各 R_i のタプル数を t_i $(=|R_i|)$，A_i のイメージサイズを d_i で表す．R_i と R_0 の接合選択率を

$$s_i = \frac{|R_i \bowtie R_0|}{t_i t_0}$$

とする．また ADT 選択 $\sigma_{f(B)}(R_0)$ に対する選択率を

$$I_f = \frac{|\sigma_{f(B)}(R_0)|}{t_0}$$

で表す．

3.2 仮　　定

ここでコスト式を定めるために必要な仮定を挙げる．類似の仮定は多くの文献にみられる[4),6),9)]．

1) 問合せは星型問合せに限定する．
2) ADT 選択に対する索引は用いない．
3) データベースは主記憶上にあるものとする．準接合，ADT 選択，接合に対するコスト式は主記憶上での処理コストに基づく．また主記憶の大きさに制限は設けない．
4) どんな準接合演算 $R_0 \ltimes R_i$，接合演算 $R_0 \bowtie R_i$ も，R_i 以外の関係と R_0 の間の接合選択率に

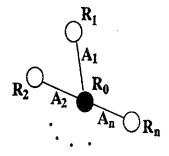


図 1 ADT 選択を含む星型問合せ
Fig. 1 Star-query with ADT selection.

変化を与えない．また ADT に対する選択率 I_f に関しても同様である．

5) 接合属性 A_i $(1\leq i\leq n)$ において各値は同一の確からしさで出現するとする．

6) 接合演算 $R_0\bowtie R_i$ のタプルの照合のためのコストは $t_0g(t_i)$ の形式とする．ここで $g(t_i)$ は関係 R_0 の1タプル当り必要なコストである．ネスト型ループ接合法，ハッシュ接合法に関してこの式が成立する．

7) 準接合演算 $R_0\ltimes R_i$ に関しては，R_i の接合属性に関して射影をとり，その結果と R_0 とを接合することによって実現されると仮定する．射影コストはソートに対するコスト $t_i\log_2(t_i)$ で見積もる．後半の接合コストは $t_0g(d_i)$ の式で表される．

8) 実行コストの算出に必要とされるパラメータ，すなわち各関係のタプル数，コスト関数 g，接合選択率，イメージサイズ，ADT選択率等は，前もって与えられているとする．

3.3 2つの関係のケース

最も単純な星型問合せである2つの関係のケース，すなわち R_0 が ADT 関係であり，R_1 が ADT 選択に寄与しない通常関係であるケースについての問合せ実行プラン (QEP) とコスト式を分析し，一般の n 個の関係に対するヒューリスティックを導く．

2章で述べたように2つの関係のケースに対しては以下の3つの QEP が考えられる．

QEP 1 : $\sigma_f(R_0)\bowtie R_1$

QEP 2 : $\sigma_f(R_0\bowtie R_1)$

QEP 3 : $\sigma_f(R_0\ltimes R_1)\bowtie R_1$

QEP 1, QEP 2, QEP 3 の各コストを C_1, C_2, C_3 で表すと，これらは以下のように表される．

$$C_1=\lambda t_0+I_f t_0 g(t_1)$$

$$C_2=t_0g(t_1)+\lambda t_0t_1s_1$$

$$C_3=t_1\log_2(t_1)+t_0g(d_1)+\lambda t_0d_1s_1+I_f t_0d_1s_1g(t_1)$$

ここで λ は1タプル当りの ADT 選択演算コストである．C_1 は ADT 選択コスト λt_0 と選択後の接合コスト $I_f t_0g(t_1)$ の和として表される．C_2 は接合コスト $t_0g(t_1)$ と接合後の関係に対する ADT 選択コスト $\lambda t_0t_1s_1$ の和として表される．また C_3 の第一，第二項は準接合コストを，第三項はその後の ADT 選択コストを，最後の項は接合コストを表す．

C_3 の第三項，つまり ADT 選択コストの項は自明ではないので簡単な説明を以下に述べる．射影後の関係 R_1 を $R_1'=\Pi_{A1}(R_1)$ とする．このとき明らかに $|R_1'|=d_1$，また $t_1/d_1=k_1$ とおく．仮定で A_1 において各値が同一の確からしさで出現することから

$$|R_0\bowtie R_1'|=|R_0\bowtie R_1|/k_1$$

となる．R_1' と R_0 との接合選択率

$$s_1'=\frac{|R_0\bowtie R_1'|}{|R_0||R_1'|}=\frac{|R_0\bowtie R_1|/k_1}{|R_0|d_1}$$

$$=\frac{t_0t_1s_1/k_1}{t_0t_1/k_1}=s_1$$

となる．したがって ADT 選択コストは

$$\lambda t_0d_1s_1'=\lambda t_0d_1s_1$$

となる．

ADT 選択コスト λt_0 が接合コスト $t_0g(t_1)$ より十分に小さいとき，すなわち $\lambda\ll g(t_1)$ が成立するとき QEP 1 が最も有利であることは明らかである．$\lambda\gg g(t_1)$ が成立するときは各コスト式の主要な項は λ を含んだ項になる．すなわち各コストは以下のように近似できる．

$$C_1\approx\lambda t_0$$

$$C_2\approx\lambda t_0t_1s_1$$

$$C_3\approx\lambda t_0d_1s_1$$

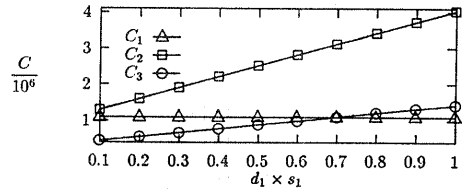
$t_1\geq d_1$ より $C_3<C_2$．また接合属性 A_1 において各値が同一の確からしさで出現するとすると $d_1s_1\leq 1$ が成立する (付録参照)．したがって $C_3\leq C_1$ が成立する．以上のことは ADT 選択コストが接合コストより十分に大きいときには準接合を先に実行する QEP 3 が最も有利であることを示している．

ADT 選択コストと接合コストが比較的近いケースについてコスト式の見積もりを行った結果が**図2**(a)(b)，$\lambda\gg g(t_1)$ のケースの例が図2(c)，$\lambda\ll g(t_1)$ のケースの例が図2(d)に示されている．これらの見積もりではいずれも $t_0=t_1=1000$，$I_f=0.1$ が仮定されている．また接合演算としてネスト型ループを仮定し $g(t_1)=t_1$，$g(d_1)=d_1$ としてある．

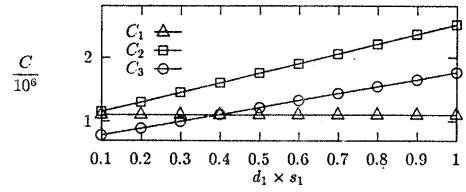
$\lambda\gg g(t_1)$ のケースでは $s_1d_1\leq 1$ の全領域で $C_3\leq C_1$，$C_3\leq C_2$ が成立している．Yajima らによる QEP 2 が $s_1>1$ (すなわち $s_1d_1>0.6$) で $C_2>C_1$ になるのを考えると QEP 3 がこのケースでは明らかに有利である．

$\lambda\ll g(t_1)$ のケースでは $C_1\leq C_2$，$C_1\leq C_3$ が成立している．この例は $d_1\approx t_1$ の場合で，われわれのシミュレーションではこの状況 (すなわち $\lambda\ll g(t_1)$，$d_1\approx t_1$) に限り $C_2\leq C_3$ が成立する結果が得られている．

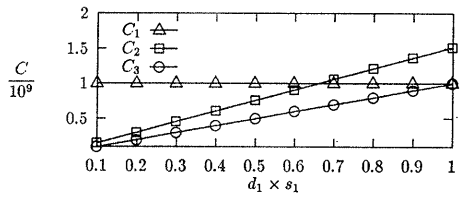
$\lambda\approx g(t_1)$ のケースでは C_2，C_1 は d_1 によらないが



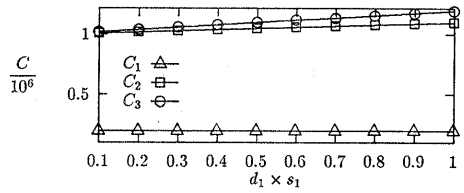
(a) d_1=330, λ=1000



(b) d_1=660, λ=1000



(c) d_1=660, λ=1000000



(d) d_1=990, λ=100

図 2　QEP の比較
Fig. 2　QEP comparison.

C_3 は d_1 が小さくなるほどコストが小さくなり有利なことがわかる．

以上の結果から以下のことが示される．

1. λ が大きいほど QEP 2, QEP 3 が有利になる．特に s_1d_1 の全領域で $C_3 \leq C_1$ が成立する点 QEP 3 は有利である．
2. s_1d_1 が小さいほど QEP 3 が有利になる．特に QEP 3 は d_1 が小さいときほど有利である．
3. 接合，あるいは準接合による ADT 削減効果が得られるケースでは $C_3 \leq C_2$ が成立し QEP 3 が QEP 2 よりも有利である．

この結果は，QEP 1 と QEP 3 のみを考慮すれば広いパラメータの範囲で最適の QEP が得られることを示している．この 2 関係の分析に基づき一般の星型問合せにおいても QEP 3 型の準接合を先行する実行プランと従来の選択を先行する実行プラン（これを準接合を全く行わない QEP 3 型の実行プランとみなす）のみを QEP の探索空間とする．

3.4 (n+1) 個の関係に対するコスト式

この節では中心ノードと n 個の葉ノードをもつ星型問合せに対する次の QEP に対するコスト式を導出する．まず $k \leq n$ 個の関係 $R_{i_1}, \cdots, R_{i_k}$ に関する準接合列を実行する．

$$R_0 := R_0 \ltimes R_{i_1}, \cdots, R_0 := R_0 \ltimes R_{i_k}$$

次に R_0 に対する ADT 選択

$$R_0 := \sigma_f(R_0)$$

を実行し，最後に n 個の関係との接合

$$R_{j_1} \bowtie (\cdots (R_{j_i} \bowtie (\cdots \bowtie (R_{j_n} \bowtie (R_0)) \cdots)$$

を実行する．すなわち準接合に関与する k 個の関係とその準接合列，および n 個の関係の接合の線形順序が決定されたとき，準接合，ADT 選択，接合を順次実行する QEP に対するコスト式を考える．全コストを $Cost_t$，準接合コストを $Cost_s$，ADT 選択コストを $Cost_A$，接合コストを $Cost_j$ で表す．$Cost_t$ は以下で表される．

$$Cost_t = Cost_s + Cost_A + Cost_j$$

一般性を失うことなく準接合に関与する関係列を $(R_1, \cdots, R_k)$ で表す．準接合コストは以下のように表される．

$$\begin{aligned} Cost_s &= t_1 \log_2(t_1) + t_0 g(d_1) \cdots \\ &\quad + t_k \log_2(t_k) + t_0 (d_1 s_1 \cdots d_{k-1} s_{k-1}) g(d_k) \\ &= \sum_{i=1}^{k} t_i \log_2(t_i) + \sum_{i=1}^{k} t_0 g(d_i) \prod_{j=1}^{i-1} d_j s_j \end{aligned}$$

$R_1, \cdots, R_k$ の関係に対する準接合列が実行された後の関係 R_0 のタプル数は $t_0 d_1 s_1 \cdots d_k s_k$ であることより

$$Cost_A = \lambda t_0 (d_1 s_1 \cdots d_k s_k) = \lambda t_0 \prod_{i=1}^{k} d_i s_i.$$

準接合，ADT 選択が終了した後は接合演算が残る．接合演算の最適順序決定問題は各種ヒューリスティックに基づく多くの提案がなされている．ここでは接合コストを接合に関与する各関係のタプル数，各関係の接合選択率に対する以下の関数として与える

$$JC(T_0, T_1, \cdots, T_n, S_1, \cdots, S_n).$$

ここで T_0, T_i, S_i $(1 \leq i \leq n)$ はおのおの中心ノードのタプル数，葉ノードのタプル数，葉ノードの接合選択率を表すパラメータとする．いま $R_1, \cdots, R_k$ までの準接合列，ADT 選択を実行した後のパラメータは次のようになる．

$$\begin{cases} T_0 = I_f t_0 \prod_{i=1}^{k} d_i s_i \\ T_i = t_i \quad 1 \le i \le n \\ S_i = \begin{cases} 1/d_i & 1 \le i \le k \\ s_i & \text{otherwise} \end{cases} \end{cases}$$

上のパラメータ S_i の値に関しては自明ではないので，以下に簡単に述べる．接合属性 A_i の各値は同一の確からしさで出現すると仮定する．重複度 k_i を

$$k_i = t_i/d_i$$

で定義する．また $R_0' = R_0 \times R_i$，$t_0' = |R_0'|$ とする．以下の関係は明らかである．

$$|R_0'| = \frac{|R_0 \bowtie R_i|}{k_i}$$

これを用いると

$$s_i = \frac{|R_0' \bowtie R_i|}{|R_0'|t_i} = \frac{|R_0 \bowtie R_i|}{|R_0'|t_i}$$

$$= \frac{|R_0 \bowtie R_i| k_i}{|R_0 \bowtie R_i| t_i} = 1/d_i$$

が求められる．

上記の議論をまとめると与えられた QEP に対する全コストは

$$Cost_t = \sum_{i=1}^{k} t_i \log_2(t_i) + \sum_{i=1}^{k} t_0 g(d_i) \prod_{j=1}^{i-1} d_j s_j + \lambda t_0 \prod_{i=1}^{k} d_i s_i + JC(I_f t_0 \prod_{i=1}^{k} d_i s_i, t_1, \cdots, t_n, 1/d_1, \cdots, 1/d_k, \cdots, s_n) \quad (3.1)$$

で表される．

4. 最適化

この章ではまず与えられた問合せに対する最適な QEP を与えるアルゴリズムについて述べる．続いて問合せに関与する関係の数が多い場合に有効な近似アルゴリズムを示し，幾つかの例に対して適用する．

4.1 最適アルゴリズム

この論文で対象とする QEP は，3.4 節でコスト式を与えた準接合，ADT 選択，接合を順次実行するものである．準接合に関与する関係の集合（準接合集合）S が与えられたとき最適 QEP はコスト式(3.1)を最小にする準接合系列および接合系列を求めることにより得られる．このときの QEP を optQEP(S)，最適コストを minCost(S) で表す．ここで述べる最適アルゴリズムは葉ノードの集合 $\mathcal{R} = \{R_1, \cdots, R_n\}$ が与えられたとき

$$Min\{minCost(S) | S \in 2^{\mathcal{R}}\}$$

となる optQEP(S) を決定する．$|\mathcal{R}| = n$ とすると考察対象となる QEP の数は

$$n! \sum_{i=1}^{n} {}_nP_i = (n!)^2 \sum_{i=1}^{n} \frac{1}{(n-i)!} \approx (n!)^2 e$$

で与えられる．すなわち各準接合集合の各準接合列に対して $n!$ 個の接合列が存在する．

コスト式(3.1)は第2項が準接合列に，第4項の計算は接合列におのおの独立に依存する．第1，第3項は一定とみなせる．したがってコスト式(3.1)は第2項を最小とする準接合列と第4項を最小にする接合列をもつ QEP によって最小となる．(3.1)式の第2項を

$$C(S) = t_0 \sum_{i=1}^{k} g(d_i) \prod_{j=1}^{i-1} d_j s_j$$

で表す．ここで列 $S = (i_1 \cdots i_k)$ は列 $(1 \cdots k)$ の一つの置換を表す．このとき $C(S)$ は次のように定義することができる．

$C(\wedge) = 0$; 空列に対して
$C((i)) = t_0 g(d_i) (i = 1, \cdots, k)$; 列（i）に対して
$C(S_1 S_2) = C(S_1) + T(S_1) * C(S_2)$; 任意の部分列 S_1, S_2 に対して

ここで $T(S)$ は次のように与えられている．

$T(\wedge) = 1$; 空列に対して
$T(S) = \prod_{i \in S} (s_i d_i)$; 任意の列に対して

上記の定義は，$C(S)$ が文献 6)，7) 等の接合コスト評価式で指摘された隣接交換性（ASI 特性）を満足することを示している．すなわち，任意の非空列 S に対して rank 関数を

$$rank(S) = \frac{1 - T(S)}{C(S)}$$

と定義したとき $C(S)$ は次のような性質をもつ．任意の列 A，B と非空の列 U，V が与えられたとき

$$C(AUVB) \le C(AVUB)$$

が成立するための必要十分条件は

$$rank(U) \ge rank(V)$$

である．

この結果から導かれる結論は，$C(S)$ を最小にする S の順序は S 中の各関係 R_i に対する rank 関数

$$rank(i) = \frac{1 - s_i d_i}{t_0 g(d_i)}$$

の値の降順に関係を並べることによって得られることである．

第4項を求める方法，すなわち最適接合順序を求める方法に対しては多くの提案があるが，コスト見積もりとして具体的値が必要な場合，LPT (Linear Processing Tree) に基づく非巡回グラフに対する最適ア

ルゴリズムとして知られる KBZ 法[6]を用いることを仮定する．

準接合集合 S が与えられたとき minCost(S) を決定する手段は上で与えられた．最適 QEP は全ての $S\in 2^{\mathcal{R}}$ に対して minCost(S) を求めその最小値を求めることにより決定される．本方法は明らかに最適 QEP を与えるがその計算量は $O(2^{|\mathcal{R}|})$ となり，問合せに関与する関係の数が大きくなると実用性を失う．次に関係数が大きくなったときに有効な近似アルゴリズムについて述べる．

4.2 近似アルゴリズム

この節では関係の数の平方に比例する近似アルゴリズムと，幾つかの例に対する適用について述べる．

この近似では，準接合関係の集合 S をヒューリスティックを用いて決定し，前節の最適化手法に基づき minCost(S) を計算し，そのときの QEP を求める．

準接合を行わない場合の QEP を QEP_0 とする．すなわち QEP_0: $\sigma_f(R_0)$; $R_{i_1}\bowtie(R_{i_2}\bowtie(\cdots(R_0))\cdots)$
ここで $R_{i_1},\cdots,R_{i_n}$ は $\sigma_f(R_0)$ を実行した後の最適接合列である．葉ノード R_i に対し，次の QEP を考える．

$QEP(R_i)$: $R_0\ltimes R_i$; $\sigma_f(R_0)$;
$R_{j1}\bowtie(R_{j_2}\bowtie(\cdots(R_0))\cdots)$

すなわち $QEP(R_i)$ は R_i に関する準接合のみを行った場合の QEP である．QEP_0 と $QEP(R_i)$ のコストをそれぞれ $Cost_0$, $Cost_i$ とすると

$$Cost_0=\lambda t_0+JC(I_f t_0, t_1,\cdots,t_n,s_1,\cdots,s_n)$$
$$Cost_i=t_i\log_2 t_i+t_0 g(d_i)+\lambda t_0 d_1 s_1 + JC(I_f t_0 d_i s_i, t_1,\cdots,t_n,s_1,\cdots,1/d_i,\cdots,s_n)$$

と表される．R_i が

$Cost_i\leq Cost_0$

を満足するとき，R_i を準接合に関して有効であるという．これに基づき各 R_i が準接合に関与するか否かを他の関係と独立に決定する．ここで準接合に関して有効な関係の集合 S' を

$S'=\{R_i\mid R_i$ は準接合に関して有効である$\}$

とする．近似アルゴリズムでは，準接合集合としてこの S' のみを考える．

上記ヒューリスティックに基づく近似アルゴリズムは以下のとおりになる．

ステップ1：各葉ノードが準接合に関して有効であるか否か決定し，準接合に関与する集合 S' を求める．
ステップ2：集合 $S'=\{R_1,\cdots,R_k\}$ 中の関係をその rank の降順 $(R_{i_1},\cdots,R_{i_k})$ に並べ，それに従って準接合列

$R_0:=R_0\ltimes R_{i_1};\cdots;R_0:=R_0\ltimes R_{i_k}$

を生成する．ADT 選択演算を付加する $\sigma_f(R_0)$
ステップ3：特定のアルゴリズムに基づき接合順序を決定する．

上記のアルゴリズムに関して JC の計算に KBZ 法を用いると仮定する．星型問合せに関しては葉ノードの数を n とすると，JC の計算コストは $O(n)$ である．ステップ1の各 R_i に対する計算コストは $O(n)$ となるのでステップ1全体としては $O(n^2)$ となる．ステップ2に関しては最悪のケースで $O(n\ln n)$ である．またステップ3は $O(n\ln n)$ であり全体としては $O(n^2)$ となる．

4.3 適 用 例

この節では前述のアルゴリズムを幾つかの例に適用した結果を示す．コスト式の適用に関してはネスト型ループ接合を仮定した．したがって $g(t_1)=t_1$, $g(d_1)=d_1$ が仮定される．また $t_0=1000$, $\lambda=10^5$, $I_f=0.3$ を仮定する．**表3**に葉ノードが4つのケースについての例を示した．この結果は最適アルゴリズムと近似アルゴリズムが一致している．この場合，全コストが従来の1/5程度に減少していることがわかる．**表4**に葉ノードが8個のケースについての例を示した．この例では最適アルゴリズムと近似アルゴリズムの間では関係 R_2 の準接合を行うかどうかの相違がある．この2つの実行プランに対する全コストに関してはほとんど有意の差が認められない．いずれのコストも従来の方法に比べ 1/10 程度までコストを減少させていることがわかる．準接合集合の選び方の全コストに対する影響を調べるため，通常関係を rank の順に1つずつ加えてできる集合列，すなわち

表3 最適化例 ($n=4$)
Table 3 An example ($n=4$) of optimization.

R_i	t_i	s_i	d_i	$Cost_i-Cost_0$	有効
R_1	1000	0.00117	850	3.04×10^5	×
R_2	950	0.00111	900	8.08×10^5	×
R_3	1300	0.00070	980	-3.06×10^7	○
R_4	1400	0.00050	700	-6.48×10^7	○

	準接合列	全コスト
従来の QEP		1.0107×10^8
最適アルゴリズムで得られた QEP	R_4R_3	2.5506×10^7
近似アルゴリズムで得られた QEP	R_4R_3	2.5506×10^7

表 4 最適化例 (n=8)
Table 4 An example (n=8) of optimization.

R_i	t_i	s_i	d_i	$Cost_i - Cost_0$	有効
R_1	1000	0.00117	850	3.04×10^5	×
R_2	1000	0.00998	100	-9.37×10^4	○
R_3	950	0.00111	900	8.08×10^5	×
R_4	900	0.00080	700	-4.36×10^7	○
R_5	1300	0.00070	980	-3.06×10^7	○
R_6	1400	0.00050	700	-6.49×10^5	○
R_7	850	0.00150	570	-1.40×10^7	○
R_8	1120	0.00170	400	-3.20×10^7	○

	準接合列	全コスト
従来の QEP		1.0184×10^8
最適アルゴリズムで得られた QEP	$R_6R_8R_4R_5R_7$	9.9042×10^6
近似アルゴリズムで得られた QEP	$R_6R_8R_4R_5R_7R_2$	9.9046×10^6

$\emptyset, \{R_{1'}\}, \{R_{1'}, R_{2'}\}, \cdots, \{R_{1'}, \cdots, R_{8'}\}$
に対しておのおのを準接合集合 S としたときの min Cost(S) が図 3 にプロットされている．この図ではいちばん左側の点が準接合を全く行わない場合のトータルコスト，すなわち従来の問合せ手法で生成される QEP に対するコスト，を表し，5，6番目の部分列に対応する点はそれぞれ最適アルゴリズムと近似アルゴリズムにより得られたトータルコストを表す．この例は最初の数個の関係に対する準接合が全コストの減少に大きく寄与し，その後の関係の準接合は大きな影響を与えないことを示している．したがって準接合集合Sを近似的に決定しても全コストに大きな誤差を生じることがないことが推定される．

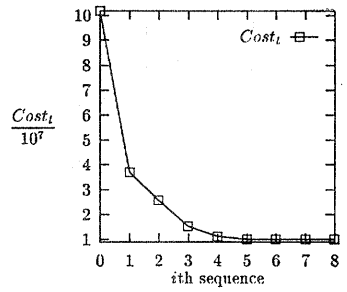


図 3 準接合列と全コストの関係
Fig. 3 Relationship between the total cost and the semijoin sequence.

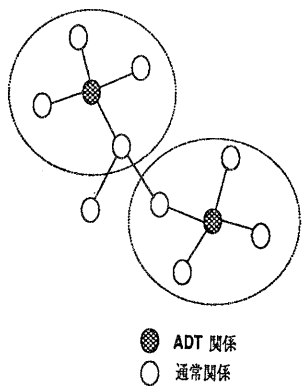


図 4 一般的な問合せグラフ
Fig. 4 Query graph in general case.

4.4 一般的問合せグラフに対する考察

本最適化アルゴリズムは ADT 関係がただ 1 つの場合の問合せのみを対象としている．図 4 のような ADT 関係を複数もつ一般的な問合せグラフで表される問合せに対しては直接的な適用はできない．しかしながらグラフ中で点線で示された部分グラフは本論文で扱っている星型問合せとみなすことができる．このようなケースでは各部分グラフに対して前節の近似アルゴリズムを適用し，準接合集合を決定し最適の準接合列を生成する．これにより ADT 選択に必要なコストを減少させ，その後は従来の最適化手法を適用するということで全コストの大幅な削減が可能となる．この問題をより詳細に検討するためには，2つの ADT 関係が直接接合されている場合に対するコスト評価を明らかにする必要がある．

5. ま と め

コストの高い ADT 選択を含む問合せ処理では ADT 関数の評価回数を減らすことが最適化の主要な目標となる．本論文では，準接合演算を用いてこの目標を達成する最適化方式を提案した．この方式は，選択—射影—接合問合せ処理の前段階で準接合演算を行い，その後従来の最適化技法と同様の選択，接合を順次実行する．この戦略に基づき，星型問合せに対しコスト最小の実行プランを生成するアルゴリズムを示した．さらに問合せに関与する関係の数が大きくなったときに有効な近似アルゴリズムを与えた．これらのアルゴリズムを幾つかの例に適用した結果，コストの高い ADT 選択を含む問合せに対しては，従来の最適化技法に比べ格段に優れた実行プランを生成することが示された．また近似アルゴリズムも最適解に十分近い解を与えることも示された．

本最適化方式は以下の特徴がある．

1. ADT 選択コストが接合コストより大きいほど有効である．

2. 接合選択率が小さいほど有効である.
3. 接合属性のイメージサイズが小さい,あるいは値の重複度が大きいほど有効である.

本論文では,問合せの形式を星型問合せに限定し,コストモデルにおいても主記憶モデルを用い,I/O コストを考慮しない.今後より一般の問合せに対する検討,ディスク上のデータベースに対するコストモデルの開発を行うことが必要である.

謝辞 本稿の内容に関し多くの有益な助言をいただいた筑波大学電子情報工学系北川博之助教授に感謝いたします.

参考文献

1) Jiang, S., Kitagawa, H., Ohbo, N. and Suzuki, I.: Abstract Data Types in Graphics Databases, *Proceedings of the IFIP TC 2/WG 2.6 Working Conference on Visual Database Systems*, pp. 239-255, Tokyo, Japan (1989).
2) Osborn, S. and Heaven, T.: The Design of a Relational Database System with Abstract Data Types for Domains, *ACM TODS*, Vol. 11, No. 3, pp. 357-373 (1986).
3) Stonebrake, M.: Inclusion of New Types in Relational Data Base Systems, *Proc. 2nd Conf. on Data Engineering*, Feb., Los Angels, CA (1986).
4) Yajima, K., Kitagawa, H., Yamaguchi, K., Ohbo, N. and Fujiwara, Y.: Optimization of Queries Including ADT Functions, *DASFAA '91*, Tokyo, Japan (1991).
5) Chen, H., Yu, X., Yamaguchi, K., Kitagawa, H., Ohbo, N. and Fujiwara, Y.: Decomposition—An Approach for Optimizing Queries Including ADT Functions, *Information Processing Letters*, Vol. 43, No. 6, pp. 327-333 (Oct. 1992).
6) Krishnamurthy, R., Boral, H. and Zaniolo, C.: Optimization of Nonrecursive Queries, *Proc. 12th Int. Conf. VLDB*, pp. 128-137, Kyoto (1986).
7) Ibaraki, T. and Kameda, T.: Optimal Nesting for Computing N-relational Joins, *TODS*, Vol. 9, No. 3 pp. 482-502 (1984).
8) Wong, E. and Ioannidis, Y.: Query Optimization by Simulated Annealing, *Proc. ACM-SIGMOD Conf. on Management of Data*, pp. 9-21 (1987).
9) Swami, A.: Optimization of Large Join Queries, *Proc. ACM-SIGMOD Conf. on Management of Data*, pp. 8-17, Chicago, Illinois (1988).
10) Carey, M., Dewitt, D. and Vandenber, S. L.: A Data Model and Query Language for EXODUS, *Proc. ACM-SIGMOD Conf. on Management of Data*, June, pp. 413-423, Chicago (1988).
11) Linch, C. and Stonebraker, M.: Extended Userdefined Indexing with Application to Textual Databases, *Proc. 14th Int. Conf. VLDB*, pp. 306-317, Los Angels, CA (1988).
12) Stonebraker, M., Rowe, L. and Hirohama, M.: The Implementation of POSTGRES, *IEEE Trans. on Knowledge and Data Engineering*, Vol. 2, No. 1, pp. 125-142 (Mar. 1990).

付　録

R_0 と任意の関係 R_i においてその接合属性 $R_0.A_i$ および $R_i.A_i$ の中で各値が同一の確からしさで出現すると仮定すると不等式 $d_{is_i} \leq 1$ が成立する.

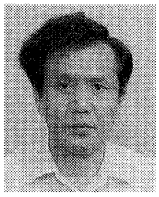
[証明] 各値の重複度を表す k_0, k_i を $k_0 = t_0/d_0, k_i = t_i/d_i$ とおく.

$$s_{0i} = \frac{|R_0 \bowtie R_i|}{|R_0 \| R_i|} = \frac{d_{0i} k_0 k_i}{d_0 k_0 d_i k_i} = \frac{d_{0i}}{d_0 d_i}$$

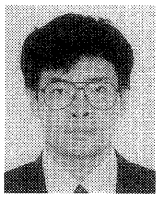
ここで d_{0i} は $R_0 \bowtie R_i$ の接合属性 A_i のイメージサイズを表す.明らかに $d_{0i} \leq min(d_0, d_i)$, したがって

$$d_i s_{0i} = \frac{d_{0i}}{d_i} \leq 1 \quad \blacksquare$$

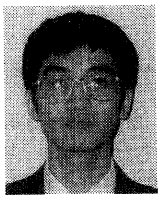
(平成5年4月5日受付)
(平成5年10月14日採録)



大保　信夫（正会員）
昭和20年6月21日生.東京大学理学部卒業.理学博士.筑波大学電子情報工学系勤務.研究テーマ:データベースシステム.ACM, IEEE-CS 学会各会員.



張　暁冬（正会員）
1963年生.1985年中国国立中山大学計算機科学系卒業.1990年筑波大学理工学研究科修士課程修了.現在同大学工学研究科博士課程在学中.研究テーマ:データベースシステム.電子情報通信学会会員.



陳　漢雄（正会員）

1964年生．1985年中国国立中山大学計算機科学系卒業．1993年筑波大学大学院工学研究科修了．工学博士．現在筑波大学電子情報工学系助手．研究テーマ：データベース，知識ベースシステム．ACM 学会会員．

藤原　譲（正会員）

1933年生．1957年東京大学工学部応用物理学科卒業．同年(株)クラレ入社．中央研究所，ノースカロライナ大学，スタンフォード大学留学を経て，1976年より筑波大学電子情報工学系教授．基礎情報学，特に情報構造解析，モデル化，データベース，情報ベース等に関する研究と応用システムの開発を行っている．電子情報通信学会，人工知能学会，情報科学技術協会，情報知識学会，ACM，IEEE，AAAI，ASIS，ACS，ASTM など各会員．