関係データベースに対するデータアクセスの数式処理による最適化について†

古 川 康 一††

近年，関係データベースの実用化を図るために，最適化に関する研究が数多くなされてきた．それらは，関係代数式上の等価変換によるものと，インデックスや分類などのデータ表現を利用するより論理的なレベルの低い処理に分けられる．

本論文では，データ表現を利用する最適化の一部を，より形式的に関係代数式上の等価変換によって実現する．データ表現としては，階層表現，多値従属による関係の分解，およびデータの冗長表現を考察の対象とする．階層表現は，インデックスの役割を果すものであるが，本論文では，2段の階層表現上で，階層構造に沿った検索手続きが，関係代数式の変換操作によって得られることを示す．また，多値従属によって分解された関係に対するある種の除算が，分数の約分に相当する変換操作によって簡単化されることを示す．最後に，データの冗長表現を利用した最適化も，同様の変換操作によって実現され，それが定理の証明における lemma の利用に相当していることを示す．

1. はじめに

近年，関係データベースの実用化を図るために，最適化が注目を浴び，それに関する研究が数多くなされてきた．それらは，手法によって分けると，関係代数式上の等価変換に基づく大局的な方法と，インデックスや分類などのデータ表現を利用する，より論理的なレベルの低い方法とがある．

前者の例としては，Smith 他[10]，および Hall[7]の研究がある．彼等は，関係代数式の等価変換規則と，各規則を適用すべきかどうかを示す条件を与え，質問式をより効率の良いものに変換してゆく．ある質問の答を出す手続きは，一意的には定まらない．たとえば，選択演算と結合演算は，その順序を入れかえても結果は同じである．しかし，効率の面では大きく異なる．すなわち，選択演算などのレコード数を減少させる演算をできるだけ先に実行した方が良い．彼等の最適化は，このような事実を用いて行われる．

後者の例としては，Astrahan 他[2]の研究が代表的なものである．彼等は，索引の利用に着目した．質問が与えられると，その各要素を索引が利用できるものと利用できないものにタイプ分けし，索引を有効に用いる手続きを作り出す．彼等は，ある場合には，一時的に索引を作り出すことも行っている．

この2つの方法は独立しているわけではなく，Smith 等は，利用できるインデックスがあるかどうかを変換規則適用の条件に組み入れている．

本論文では，データ表現を利用する最適化のうちのある部分を，より形式的に関係代数式上の等価変換によって実現することを試みる．すなわち，データ表現自身を関係代数式によって表わし，その式を用いて，データ表現に合った検索手続きの導出を，関係代数式の操作により行う．データ表現としては，階層表現，多値従属による関係の分解，および任意の冗長なデータ表現を考察の対象とする．

このような形式化を通して，データ処理における関係代数の新しい応用を示す．また，種々のデータ処理の技術を，代数上の概念とのアナロジで説明する．さらに，これらの形式化によって，より柔軟で，かつ効率の良い知的データベース・システムの実現が可能であることを示す．

以下，**2** では，関係および関係代数の諸演算を定義する．**3** では，階層構造を関係代数式によって定義し，その上での関係代数演算の諸性質を導き，それらの定義および諸性質を用いて，階層構造を意識しない質問表現から階層構造に沿った検索手続きを導出する過程を示す．**4** では，多値従属によって分解された関係に対するある種の除算が，分数の約分に相当する変換操作によって簡単化されることを示す．**5** では，冗長なデータ表現を利用した最適化が同様の変換操作に

† Optimizing the Data Access to a Relational Data Base by Formula Manipulation by KOICHI FURUKAWA (Information Systems Section, Computer Science Division, Electrotechnical Laboratory).

†† 電子技術総合研究所ソフトウェア部情報システム研究室

よって実現され，それが定理の証明における lemma の利用に相当していることを示す．最後に，まとめと今後の課題についてふれる．

2. 関係および関係代数

はじめに関係データベースの形式的な定義を与える．

定義 互いに識別可能な記号から成る有限集合 $\{A_1, A_2, \cdots, A_n\}$ の要素 A_i を属性という．各属性 A_i には，**定義域** D_i が付随している．

定義 属性集合 $\{A_1, A_2, \cdots, A_n\}$ 上の関係 R は，直積 $D_1 \times D_2 \times \cdots \times D_n$ の部分集合で，関係の要素 $\langle d_1, d_2, \cdots, d_n \rangle$, $d_i \in D_i$, $i=1 \sim n$, を組と呼ぶ．$\{A_1, A_2, \cdots, A_n\}$ 上の関係 R を $R(A_1, A_2, \cdots, A_n)$ と表わす．

ここで，属性 A_i の定義域を $dom(A_i)$ で表わす．また，単一の属性は A, B, …等で表わし，属性の集合（空集合も含む）を X, Y, …等で表わす．

定義 属性集合 $\{A_1, A_2, \cdots, A_n\}$ 上の関係型 $\boldsymbol{R}(A_1, A_2, \cdots, A_n)$ は，$\{A_1, A_2, \cdots, A_n\}$上のすべての関係の集合である．

関係 $R(A_1, A_2, \cdots, A_n)$ は，関係型 $\boldsymbol{R}(A_1, A_2, \cdots, A_n)$ の要素である．関係データベースの枠組を与える概念スキーマは，関係型の集合として定義される．本論文では，冗長表現，構造表現を内部スキーマとして導入するが，それも同様に関係型の集合として与えられる．

図 1 に {**EMP**（従業員，場所），**ROOM**（番号，部門，階，面積）} を概念スキーマとする関係データベースの例を示す．

つぎに関係代数の各演算を定義する[4),9)]．

定義 $R(A, X)$ を関係とするとき，R の A に関する**射影** $\Pi_A(R)$ は，次の式によって与えられる集合である．

$$\Pi_A(R) = \{\boldsymbol{x} \mid (^\exists z)(z, \boldsymbol{x}) \in R\} \tag{2.1}$$

ここで $\boldsymbol{x}$ は，属性集合 X に対応する属性値の並びを表わし，z は属性 A に対応する属性値を表わす．

定義 $R(X)$ を関係とする．$\boldsymbol{x}$を X に対応する変数とする．$f(\boldsymbol{x})$ を論理関数 $\boldsymbol{x} \to \{\mathrm{T}, \mathrm{F}\}$ とする．R の $f(X)$ に関する**選択** $\Phi_{f(X)}(R)$ は，次の式を満足する $\boldsymbol{x}$ の集合である．

$$\Phi_{f(X)}(R) = \{\boldsymbol{x} \mid \boldsymbol{x} \in R \wedge f(\boldsymbol{x}) = \mathrm{T}\} \tag{2.2}$$

選択演算は，条件 $f(\boldsymbol{x})$ による R の探索であり，連想探索を表わしている．一方，連想探索要求では，変数の値のみを求めるのが普通である．つぎの定義はそのような演算の定義である．

定義 $R(X, Y)$ を関係とする．R の $f(Y)$ に関する**自然選択演算** $\Psi_{f(Y)}(R(X, Y))$ は，つぎの式を満足する $\boldsymbol{x}$ の集合である．

$$\Psi_{f(Y)}(R(X, Y)) = \{\boldsymbol{x} \mid (^\exists \boldsymbol{y})[\langle \boldsymbol{x}, \boldsymbol{y} \rangle \in R \wedge f(\boldsymbol{y}) = \mathrm{T}]\} \tag{2.3}$$

自然選択 $\Psi_{f(Y)}$ は，射影 Π_Y と選択 $\Phi_{f(Y)}$ の合成関数 $\Pi_Y \circ \Phi_{f(Y)}$ に等しい．ここで，$\circ$ は関数の合成を表わす．

定義 $R_1(A, X)$ と $R_2(B, Y)$ を共に関係とし，$f(A, B)$ を論理関数とする．R_1 と R_2 の $f(A, B)$ に関する**結合** $R_1(A, X) \ *_{f(A,B)} \ R_2(B, Y)$ は，次の式によって与えられる組の集合である．

$$\begin{aligned} &R_1(A, X) \ *_{f(A,B)} \ R_2(B, Y) \\ &\quad = \{(a, \boldsymbol{x}, b, \boldsymbol{y}) \mid (a, \boldsymbol{x}) \in R_1 \\ &\qquad \wedge (b, \boldsymbol{y}) \in R_2 \wedge f(a, b) = \mathrm{T}\} \end{aligned} \tag{2.4}$$

結合条件が［番号＝場所］のように等式で与えられる場合，上の式で $a=b$ となり，片方は冗長である．その場合には，そのうちの一方を結果から除くことにする．これを**自然結合** (natural join) という．

定義 関係 $R_1(X, A)$ の関係 $R_2(A)$ による A に関する**商** $R_1(X, A) \ \Delta_A \ R_2(A)$ は，つぎの式を満足する集合である．

$$\begin{aligned} &R_1(X, A) \ \Delta_A \ R_2(A) \\ &\quad = \{\boldsymbol{x} \mid (^\exists y) y \in R_2(A) \\ &\qquad \wedge (^\forall y)[y \in R_2(A) \to (\boldsymbol{x}, y) \in R_1(X, A)]\} \end{aligned} \tag{2.5}$$

3. 階層表現上の検索手続きへの変換

図 1 に示した関係データベースは，**図 2** に示すよう

EMP（従業員，場所）

従業員	場所
SMITH	101
JONES	102
BROWN	101
HENRY	301
NELSON	102
MURPHY	201
LONG	202
MORGAN	201
LEE	202

ROOM（番号，部門，面積）

番号	部門	面積
101	INFO	95
102	ARCH	85
201	LANG	95
202	INFO	85
301	ARCH	60

図 1 関係データベースの例

Fig. 1 An example relational data base.

な構造を持っている．ここで矢印は，1対多の関係を表わしている．このような図式をバックマン図式という[3]．この1対多の関係は階層関係を表わし，階層データベースでは，この矢印がアクセス・パスとなる．すなわち，階層関係の上位にあるデータを指定して，それにつながる下位のデータ集合を取り出す．

いま，図2の矢印aに注目する．この矢印は，部門情報から部屋情報（番号，面積）がアクセスできることを示している．すなわち，**図3**に示す木構造において，与えられた部門をセレクタとして持つ枝を選び出して，下にたどることになる．この木の枝には，その部門に属する部屋の（番号，面積）のすべての組が置かれている．いま，この各枝に置かれている（番号，面積）の組の集合を，それぞれ別の関係と考え，それを元の関係の**副関係**と呼ぶことにする．

部門$=a_i$ の下に置かれている副関係を ROOM.a_i(番号，面積)とすると，この関係は，自然選択演算によってつぎのように与えられる．

$$\mathrm{ROOM}.a_i(\text{番号，面積}) = \Psi_{[\text{部門}=a_i]}(\mathrm{ROOM}(\text{番号，部門，面積})) \qquad (3.1)$$

(3.1)式は，a_i を変数と考えれば，任意の枝に置かれている関係の定義式となっている．

つぎに，関係 EMP についての同様な階層化を行う．場所 b_j に勤務している従業員より成る関係を EMP.b_j(従業員) と表わすと，それは

$$\mathrm{EMP}.b_j(\text{従業員}) = \Psi_{[\text{場所}=b_j]}(\mathrm{EMP}(\text{従業員，場所})) \qquad (3.2)$$

と定義される．

このようにして得られた2つの関係の集合 $\{\mathrm{ROOM}.a_i \mid a_i \in dom(\text{部門})\}$ および $\{\mathrm{EMP}.b_j \mid b_j \in dom(\text{場所})\}$ が，{ROOM, EMP} データベースの内部スキーマを構成するものと考える．

つぎに，アクセス・パスを指定した検索が関係代数上でどのように実現されるかを見るために，質問「INFO 部門に属しているすべての従業員を求めよ」を考える．その質問は関係代数式によって，次のように表わされる．

$$Q_1 = \Pi_{\text{番号}} \circ \Psi_{[\text{部門}=\mathrm{INFO}]}(\Pi_{\text{面積}}(\mathrm{ROOM}(\text{番号,部門,面積})) *_{[\text{番号}=\text{場所}]} \mathrm{EMP}(\text{従業員,場所})) \qquad (3.3)$$

この質問に (3.1)式を適用するためには，つぎに示す自然結合と自然選択の可換定理が必要となる．

定理 3.1 $R_1(A, B, X)$ および $R_2(C, Y)$ を2つの関係とすると，

$$\Psi_{[B=a_i]}(R_1(A, B, X) *_{[A=C]} R_2(C, Y)) = (\Psi_{[B=a_i]}(R_1(A, B, X))) *_{[A=C]} R_2(C, Y) \qquad (3.4)$$

が成り立つ．

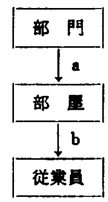

図 2 関係データベース {EMP, ROOM} に対するバックマン図式

Fig. 2 A Backman diagram for the {EMP, ROOM} relational data base.

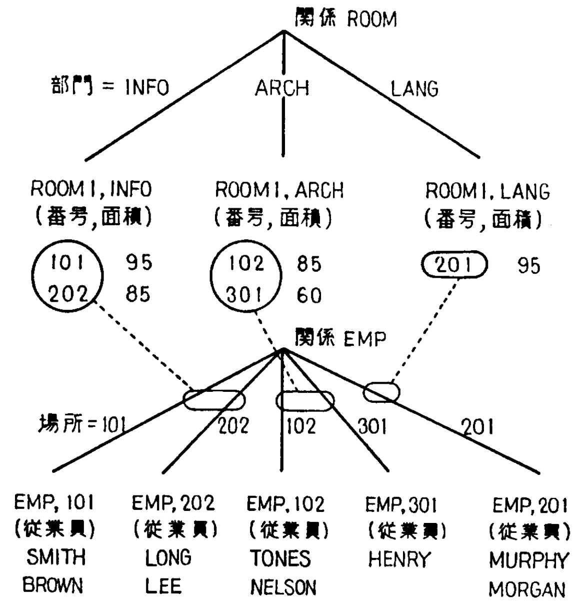

図 3 関係 ROOM および EMP の構造化

Fig. 3 A structural representation for the ROOM and EMP relations.

質問 (3.3) に定理 3.1 を適用し，その結果に，さらに定理 3.1 と同様な，射影と自然選択の可換定理を適用すると，

$$Q_1 = \Pi_{\text{番号}}(\Pi_{\text{面積}} \circ \Psi_{[\text{部門}=\mathrm{INFO}]}(\mathrm{ROOM}(\text{番号，部門，面積})) *_{[\text{番号}=\text{場所}]} \mathrm{EMP}(\text{従業員，場所})) \qquad (3.5)$$

を得る．この式の Ψ 以下に (3.1)式の左辺を代入して，

$$Q_1 = \Pi_{\text{番号}}(\Pi_{\text{面積}}(\mathrm{ROOM.\,INFO}(\text{番号，面積})), *_{[\text{番号}=\text{場所}]} \mathrm{EMP}(\text{従業員，場所})) \qquad (3.6)$$

を得る．(3.6) 式をさらに簡単化するためには，つぎ

の2つの定理が必要である.

定理 3.2 $R_1(A,X)$ および $R_2(B,Y)$ を関係とする. さらに $R_2(B,Y)$ は, B によって, 関係の集合 $\{R_2.b_j(X): b_j \in dom(B)\}$ に分解され, 階層化されているとする. そのとき

$$R_1(A,X) \quad *_{[A=B]} \quad R_2(B,Y) = \bigcup_{b_j \in \Pi_X(R_1(A,X))} (\Phi_{[A=b_j]}(R_1(A,X)) \times R_2.b_j(Y)) \tag{3.7}$$

が成り立つ. ここで×は直積である.

系 $R_1(A,X)=R_1(A)$ ならば,

$$R_1(A) \quad *_{[A=B]} \quad R_2(B,Y) = \bigcup_{b_j \in R_1(A)} (\{b_j\}(A) \times R_2.b_j(Y)) \tag{3.8}$$

が成り立つ. ここで $\{b_j\}(A)$ は, 属性が A で, ただ1つの要素 b_j のみから成る関係を表わす.

定理 3.3 $R_1(A)$ を関係とし,
$\{R_2.b_j(Y) | b_j \in dom(A)\}$ を関係の集合とすると, つぎの式が成り立つ.

$$\Pi_A \circ \bigcup_{b_j \in R_1(A)} (\{b_j\}(A) \times R_2.b_j(Y)) = \bigcup_{b_j \in R_1(A)} (R_2.b_j(Y)) \tag{3.9}$$

定理 3.2 は, ある関係 R_1 と構造化された関係 R_2 をセレクタ情報によって結合するアルゴリズムを与えている. すなわちその場合には, $[A=b_j]$ によって選択された R_1 の部分集合 $\Phi_{[A=b_j]}(R_1(A,X))$ とセレクタ b_j の下の関係 $R_2.b_j$ の直積を, R_1 の A 属性に現れる各 b_j について求め, それらの合併集合を求めればよいことを示している. また, 定理 3.3 は, そのような結合の結果に対して A 属性についての射影を行う場合には, 上に述べた直積すら必要としないことを言っている.

(3.6) 式に, 定理 3.2 の系を適用して,

$$Q_1 = \Pi_{番号} \circ \bigcup_{b_j \in R_1(番号)} (\{b_j\}(番号) \times \mathrm{EMP}.b_j(従業員)) \tag{3.10}$$

を得る. ここで,

$$R_1(番号) \equiv \Pi_{面積}(\mathrm{ROOM\,1.INFO}(番号, 面積)) \tag{3.11}$$

である. (3.10) に定理 3.3 を適用して,

$$Q_1 = \bigcup_{b_j \in R_1(番号)} \mathrm{EMP}.b_j(従業員) \tag{3.12}$$

を得る. この式を実行するには, まず, (3.11)式の R_1(番号)を求めることが必要となる. これは, 部門 INFO に属するすべての部屋の番号である. つぎにそのような各番号 b_j について, EMP. b_j(従業員), すなわち b_j に勤務しているすべての従業員を求め, それらの合併集合を求めている. この求め方は, 実は図3の木構造を上から辿って検索する方法そのものになっている.

しかしながら, この方法は階層データベースでのアクセス法と全く同じではない. 本方法では, INFO に属する部屋番号 b_j から b_j に勤務している従業員を求めるのに, 関係 EMP. b_j をアクセスしているが, この関係 EMP. b_j は, 関係 ROOM. INFO の組からポインタで指されているわけではなく, b_j 自身を手がかりとして見つけなければならない. すなわち, 関係 EMP. b_j のアクセスは, 連想探索によらなければならない. この連想探索は, ハッシングや連想メモリなどにより高速化されるが, ポインタによるアクセスには及ばない. (3.12) 式を, ポインタによるアクセスを用いた手続きに変換することが必要となる. 筆者は, 述語論理によるデータ構造の定義を用いてこの変換が可能であることを明らかにしたが, その詳細については, 別稿にゆずる[6].

質問が, 階層構造に沿っておらず, バックマン図式の矢印を下から上を辿らなければならないときには, 質問は, (3.12)式のようには簡単化されない. 例として, Smith 氏の部門を求める質問を考えると, それは,

$$Q_2 = \Pi_{番号} \circ \Psi_{[従業員=\mathrm{SMITH}]} (\Pi_{面積}(\mathrm{ROOM}(番号, 部門, 面積)) \; *_{[番号=場所]} \; \mathrm{EMP}(従業員, 場所)) \tag{3.13}$$

と表わされる. この場合, 定理 3.2 は適用できるが, 階層構造の定義式 (3.1) あるいは(3.2), および定理 3.3 は適用できず, 結局

$$Q_2 = \Pi_{番号} \circ \Psi_{[従業員=\mathrm{SMITH}]} \circ \bigcup_{b_j \in R_1'(番号)} (\Phi_{[番号=b_j]} \circ \Pi_{面積}(\mathrm{ROOM}(番号, 部門, 面積)) \times \mathrm{EMP}.b_j(従業員)) \tag{3.14}$$

となる. ここで, R_1'(番号) は, 次式で支えられる.

$$R_1'(番号) = \Pi_{部門} \cdot \Pi_{面積}(\mathrm{ROOM}(番号, 部門, 面積)) \tag{3.15}$$

(3.14) 式および (3.15) 式には, 関係 ROOM がそのままの形で残っている. ところが, 関係 ROOM は階層化され, 副関係 ROOM. a_i の集合に分解されているので, その形にすることが必要となる. そのためには, つぎのような等式を必要とする.

$$\mathrm{ROOM}(番号,\ 部門,\ 面積) = \bigcup_{a_i \in dom(部門)} \mathrm{ROOM}.a_i(番号,\ 面積) \quad (3.16)$$

(3.16) 式を (3.14) 式および (3.15)式に代入することによって，求める手続きが得られたことになる．

一方，(3.14) 式は，変換される以前の式 (3.13) に比べて，一層複雑になっている．これは，階層構造が，階層に逆らったデータのアクセス機構を持っていないからである．このような場合には，階層化されない関係表現があれば，それを利用した方がよい．そのためには，まず，内部スキーマ自身が，2つのデータ表現を持つことが必要になる．もし，物理的に異なった実体を2つの表現に合わせて用意しなければならないとすれば，この方法の長所は，打ち消されてしまう．すなわち，物理的には1つのデータ表現に対して，2つの異なるデータの見方 (view) ができる機構が必要となる．われわれが開発した関係データベース管理システム GEPHERD[12]では，同一のデータ表現に対して view ごとに異なるデータ操作関数を用意することによって，この機能を実現した．図4に，GEPHERD で採用した階層化された関係の物理表現を示す．

このように，内部スキーマに対して複数の view を与えることによって，実は初めて検索手続きの最適化の問題が生ずる．すなわち，質問に応じて，どの view を選択することによって，質問の手続きがより効率良くなるかという問題が発生する．

ここで，定理 3.2 の適用を制限することによって，(3.12)式の導出を可能にしたままで (3.14) 式が導出されないようにすることが必要となる．Q_1 と Q_2 の違いは，Q_1 では，階層構造のセレクタによる選択演算が行われており，Q_2 では行われていない点である．この条件，すなわち，階層構造のセレクタによる選択演算を含むこと，を定理 3.2 の適用条件とすればよい．

なお，ここで与えた変換規則群は，それだけで十分ではない．たとえば，階層構造の下にある関係のセレクタによる選択演算がある場合には，定理 3.2 の(3.7) 式に類似した，つぎのような変換規則が必要である．

$$R_1(A, X) \ *_{[A=B]}\ \Phi_{[B=b_i]}(R_2(B, Y)) = \Phi_{[A=b_i]}(R_1(A, X)) \times R_2.b_i(Y) \quad (3.17)$$

4. MVD 分解による division の簡約化[5]

関係が分解できるための条件の 1 つに多値従属性 (Multi-Valued Dependency, MVD) があることは，

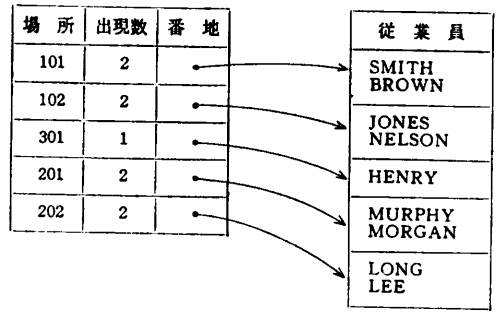

図 4 階層化された関係 EMP のデータ表現

Fig. 4 A data representation for the hierarchically structured EMP relation.

よく知られている．MVD の概念は，多少とらえ難いが，要約すれば，関係が3つの属性群 X, Y, Z に分けられ，そのうちのある1つの属性群 X からほかのいずれかの属性群への，要素——集合関係が，残りの属性群の要素をいかなる実現値に1つ固定しても変らないという従属関係である．このとき，$X \rightarrow\rightarrow Y$ のように書く ($X \rightarrow\rightarrow Z$ も同時に成り立つ)．多値従属性 $X \rightarrow\rightarrow Y$ が成り立つとき，関係 $R(X, Y, Z)$ は，$S(X, Y) = \Pi_Z(R(X, Y, Z))$ と $T(X, Z) = \Pi_Y(R(X, Y, Z))$ に分解できる．さらに，分解された2つの関係 $S(X, Y)$ および $T(X, Z)$ を X について結合することにより，元の関係 $R(X, Y, Z)$ を復元することができる．すなわち

$$R(X, Y, Z) = S(X, Y) \ *_{[X=X]}\ T(X, Z) \quad (4.1)$$

が成り立つ．またこの条件 (4.1) が，多値従属性 $X \rightarrow\rightarrow Y$ が成り立つための必要十分条件であることも知られている．

例として，図5のような関係 SUPPLY を考えよう．この関係では，すべての業者は，部品を型単位で納入しており，型→→業者の MVD が成り立つ．したがって，この関係は図6のような2つの関係 SUPPLY TYPE と PART に分解することができる．こ

SUPPLY (業者, 部品, 型)

業者	部品	型
{A パーツ, B 工業}	{プラグ, シリンダ, ピストン}	E
{C 塗装, D ペイント}	{シンナー, 塗料, ハケ}	P
{E 製鋼, F 製鉄}	{鋼板, 車軸}	I

図 5 型→→業者の MVD が成り立つ関係 SUPPLY の例

Fig. 5 A SUPPLY relation with a type →→ supplier MVD.

SUPPLYTYPE (業者,	型)
Aパーツ	E
B 工 業	E
C 塗 装	P
E 製 鋼	I
F 製 鉄	I
Dペイント	P

TYPE (部 品,	型)
鋼 板	I
車 軸	I
シリンダ	E
シンナー	P
塗 料	P
ハ ケ	P
ピストン	E
プラグ	E

図 6 図5の関係 SUPPLY の MVD による分解
Fig. 6 MVD decomposition of the SUPPLY relation in Fig. 5.

こで，これら2つの関係は，元の関係 SUPPLY から，次式により得られる．

$$\text{SUPPLYTYPE}(業者,\ 型) = \Pi_{部品}(\text{SUPPLY}(業者,\ 部品,\ 型)) \quad (4.2)$$

$$\text{TYPE}(部品,\ 型) = \Pi_{業者}(\text{SUPPLY}(業者,\ 部品,\ 型)) \quad (4.3)$$

これら2つの式は，変換規則として用いられる．

ここで，質問「型 A の部品をすべて納入している業者を求めよ」を考える．この質問は，関係 SUPPLY によって

$$Q_3 = \Pi_{型}(\text{SUPPLY}(業者,\ 部品,\ 型)) \ \Delta_{部品} \ \Psi_{[型=A]} \circ \Pi_{業者}(\text{SUPPLY}(業者,\ 部品,\ 型)) \quad (4.4)$$

と表わされる．ここで，(4.3) 式を適用して，

$$Q_3 = \Pi_{型}(\text{SUPPLY}(業者,\ 部品,\ 型)) \ \Delta_{部品} \ \Psi_{[型=A]}(\text{TYPE}(部品,\ 型)) \quad (4.5)$$

を得る．ここで，MVD 型→→業者が成り立っているので，(C, A)∈SUPPLYTYPE (業者，型) とすると，業者 C は型 A のすべての部品を納入していることになる．ゆえに，

$$Q_3 = \Psi_{[型=A]}(\text{SUPPLYTYPE}(業者,\ 型)) \quad (4.6)$$

となることが予想される．(4.5) から (4.6) への変換は，つぎの定理によってなされる．

定理 4.1 関係 $R(X, Y, Z)$ において，MVD: $Z \to\to X$ が成り立っており，$R_1(X, Z) = \Pi_Y(R(X, Y, Z))$，$R_2(Y, Z) = \Pi_X(R(X, Y, Z))$ とする．さらに，R_2 において関数従属性 $Y \to Z$ が成り立っており，かつ $\Psi_{[Y=a_i]}(R_2(Y, Z)) \neq \phi$ とする． そのとき，次の式が成り立つ．

$$\Pi_Z(R(X, Y, Z)) \ \Delta_Y \ \Psi_{[Z=a_i]}(R_2(Y, Z)) = \Psi_{[Z=a_i]}(R_1(X, Z)) \quad (4.7)$$

この定理は，分数の約分に相当している．すなわち，分子は $\Pi_Z(R(X, Y, Z))$ であるが，それは MVD の仮定により $\Pi_Z(R_1(X, Z) \ *_{[Z=Z]} \ R_2(Y, Z))$ に等しい．それを分母である $\Psi_{[Z=a_i]}(R_2(Y, Z))$ で割ることによって，$\Psi_{[Z=a_i]}(R_1(X, Z))$ が得られるわけである．この定理の証明は，Appendix で与える．

この定理を (4.5) 式に適用すれば，(4.6) 式が得られる． ここで， 定理の条件の1つである関数従属性 (FD) は，この例では， 関係 TYPE における部品→型の FD で，部品が決まればその型が一意的に決まることを意味している．このような事実は，最適化を行う問題解決機が参照できる形で，知識としてシステムに蓄えられていなければならない．これらの事実のほか，変換規則や定理なども知識の型で統一的にシステムに蓄え，利用することが可能である．それらの蓄積が知識ベースを構成し，問題解決機は PLANNER, QLISP, プロダクション・システムなどの人工知能用プログラミング・システム上に構築することが可能である．

5. 冗長表現に伴う最適化

関係モデルでは，一般には更新時に不都合が生じるのを防ぐために，データベースの正規表現を与え，データの冗長な表現を禁じている．しかしながら，たとえば分散データベースのように，利用者とデータの実体が置かれている場所が離れているような場合，頻繁に用いるデータは，利用者のいる末端局にコピーして利用した方が能率がよい．ここでは，この冗長表現を，内部スキーマ・レベルの表現として捉えることにする．それによって，関係モデルの枠組を損わずに，冗長表現を導入することが可能である．

データの冗長表現は，もとの冗長でない概念スキーマ上のデータ表現に関係代数演算を施して得られる．例として，図1のデータベースについて考える．ある利用者が，従業員と部門の関係について，多くの質問を発したいとき，その関係を一時的に作ってしまうことによって，各質問の処理が単純になる．いま，関係 EMPSECT (従業員，部門) を，関係 EMP および ROOM から，つぎの式によって求める．

$$\text{EMPSECT}(従業員,\ 部門) = \Pi_{番号}(\Pi_{面積}(\text{ROOM}(番号,\ 部門,\ 面積)) \ *_{[番号=場所]} \text{EMP}(従業員,\ 場所)) \quad (5.1)$$

この等式を， 関係 EMPSECT の定義方程式と呼ぶ．この定義方程式を知識ベースに変換規則として登録すると， 従業員と部門に関する質問は， 自動的に EMPSECT に対する探索に変換される． たとえば，**3** で扱った質問「INFO 部門に属しているすべての従業員を求めよ」は，(3.3) にまず射影と自然選択の可

換定理を施し，つぎに (5.1) 式を代入して

$$Q_1=\Psi_{[部門=\mathrm{INFO}]}(\mathrm{EMPSECT}(従業員,\ 部門)) \tag{5.2}$$

と変換される．

以上の操作は，定理の証明における lemma の利用に相当している．(5.1) 式は，基本的な関係から得られた関係 EMPSECT が，以下の処理で使えることを示している．すなわち，それは基本的な性質から導き出された lemma と同じ働きをしている．

6. おわりに

本論文では，質問式を，内部スキーマの定義方程式や関係代数の諸定理を用いることにより，内部スキーマに合った検索手続きに変換し得ることを示した．すなわち，大局的な方法での最適化の適用領域をデータ表現にまで拡げた．しかし，Smith などが考察しているキーによる分類の利用については，本論文の方式には含まれていない．実は，本論文で与えた方式で最適化を行った後で，Smith の方法を適用することは可能である．あるいは，Tanaka 他[11]による並列分類および並列探索を基にしたデータベース・マシンを利用すれば，分類の利用についての考察は不要になる．

本論文で与えた関係の階層表現は，関係の水平分解を与えていると考えられる．関係の水平分解は，関係代数演算によって一度に操作するデータ量を著しく減少させるのに役立つ．さらに，各副関係ごとにデータを局所化することにより，2次記憶へのアクセス回数も十分に小さくすることが可能である．

今後の研究課題の第1は，関係代数式の変換によって最適な手続きを得るための問題解決機の開発である．関係代数式の変換手続きは，ある時点で適用可能な規則が複数個あるのが普通であるので，一般に非決定的な手続きとなる．このような問題解決機にとって最も重要な点は，規則の適用条件をうまく与え，変換を制御することである．

第2の課題は，3でも述べたより物理的なデータ構造上の手続きへの変換である．これは，データ構造操作プログラムの自動合成の問題であり，Hansson 他[8]が与えた方式を適用することによって実現し得る見通しを得ている[6]．

謝辞 本研究を行う機会を与えていただいた石井治ソフトウェア部長，棟上昭男情報システム研究室長，ならびに，有益な討論をしていただいた東京工業大学米澤明憲博士，東京大学助教授大須賀節雄，北海道大学講師田中譲，および当所ソフトウェア部情報システム研究室の諸氏に感謝致します．

さらに査読者には適切なコメントをいただき，論文の改善に大変役立ちました．

参考文献

1) ANSI/X 3/SPARC: ANSI/X 3 /SPARC Study Group on Data Base Management Systems, Interim Report, ACM FDT, Vol. 7, No. 2 (1975).

2) Astrahan, M. M. and Chamberlin, D. D.: Implementation of a Structured English Query Language, CACM, Vol. 18, No. 10, pp. 580-587 (1975).

3) Backman, C. W.: Data Structure Diagrams, DATABASE (ACM SIGBDP Newsletter) Vol. 1, No. 2, pp. 4-10 (1969).

4) Codd, E. F.: Relational Completeness of Data Base Sublanguages, In Data Base Systems, Ed. R. Rustin, Prentice-Hall, pp. 65-98 (1972).

5) Furukawa, K.: Relational Strategies for Processing Universally Quantified Queries to Large Data Bases, Proc. Sixth IJCAI, pp. 294-299 (1979).

6) Furukawa, K.: Use of Data Representation Mapping in Automatic Generation of Data Base Access Procedures, 数理解析研究所講究録 396, pp. 21-38 (1980).

7) Hall, P. A. V.: Optimization of a Single Expression in a Relational Data Base System, IBM Journal Research and Developement, Vol. 20, No. 3, pp. 244-257 (1976).

8) Hansson, Å. and Tärnlund S. Å.: A Natural Programming Calculus, Proc. Sixth IJCAI, pp. 348-355 (1979).

9) Reiter, R.: On Closed World Data Bases, In Logic and Data Bases (Eds. H. Gallaire and J. Minker), pp. 55-76 (1978).

10) Smith, J. M. and Chang, P. T. Y.: Optimizing the Performance of a Relational Algebra Database Interface, CACM, Vol. 18, No. 10, pp. 568-579 (1975).

11) Tanaka, Y. et al.: Pipeline Searching and Sorting Modules as Components of a Data Flow Database Computer, Proc. of IFIP-80 (1980).

12) 古川：データベースの知的アクセスに関する研究，電子技術総合研究所研究報告第804号 (1979).

付録

定理 4.1 の証明．

$$S_1(X)=\Pi_Z(R(X,Y,Z))\ \Delta_Y\ \Psi_{[Z=a_i]}(R_2(Y,Z)),$$

$$S_2(X)=\Psi_{[Z=a_i]}(R_1(X,Z))$$

とおく．$S_1(X)=S_2(X)$ を証明すればよい．

（1） $S_1(X)\subset S_2(X)$ の証明．

$c\in S_1(X)$ とする．Δ_Y の定義より， すべての $d \in \Psi_{[Z=a_i]}(R_2(Y,Z))$ に対して，$\langle c,d\rangle\in\Pi_Z(R(X,Y,Z))$ が成り立つ．ゆえに，ある a_j が存在して，$\langle c,d,a_j\rangle\in R(X,Y,Z)$ となる．一方，$R_1(X,Z)$ および $R_2(Y,Z)$ の定義により，

$$\langle c,a_j\rangle\in R_1(X,Z) \tag{1}$$

$$\langle d,a_j\rangle\in R_2(Y,Z) \tag{2}$$

が成り立つ．また $d\in\Psi_{[Z=a_i]}(R_2(Y,Z))$ より，

$$\langle d,a_i\rangle\in R_2(Y,Z) \tag{3}$$

が成り立つ．ところが R_2 における FD：$Y\rightarrow Z$ の仮定と（2），（3）より，$a_j=a_i$ でなければならない．これと（1）から $c\in\Psi_{[Z=a_i]}(R_1(X,Z))=S_2(X)$ となる．ゆえに $S_1(X)\subset S_2(X)$ がいえる．

（2） $S_2(X)\subset S_1(X)$ の証明．

$c\in S_2(X)$ とすると，$\langle c,a_i\rangle\in R_1(X,Z)$ となる．また，$\Psi_{[Y=a_i]}(R_2(Y,Z))\neq\phi$ の仮定から，ある d が存在して，$\langle d,a_i\rangle\in R_2(Y,Z)$ が成り立つ．ゆえに，すべての $d\in\Psi_{[Z=a_i]}(R_2(Y,Z))$ に対して，$\langle c,d,a_i\rangle\in(R_1(X,Z)\ *_{[Z=Z]}\ R_2(Y,Z))$，すなわち，$\langle c,d\rangle\in\Pi_Z(R(X,Y,Z))$ が成り立つ．ゆえに $c\in\Pi_Z(R(X,Y,Z))\ \ \Delta_Y\ \ \Psi_{[Z=a_i]}(R_2(Y,Z))=S_1(X)$ が成り立つ．ゆえに，$S_2(X)\subset S_1(X)$．証明終．

（昭和 54 年 8 月 31 日受付）
（昭和 55 年 9 月 18 日採録）