リレーショナルデータベースシステムにおける連想検索機能

加藤　常員*・小沢　一雅**・今枝　国之助***

*岡山理科大学・日本学術振興会・特別研究員

**大阪電気通信大学・工学部・経営工学科

***岡山理科大学・工学部・電子工学科

（1989年9月30日　受理）

内容梗概

データベースシステムが身近なものになるにしたがって，柔軟な検索が要求されるようになった。本稿では，リレーショナルデータベースシステムにおける連想的な検索機能を提案する。提案する連想検索機能は，データベースに蓄えられているデータの特徴をエントロピーを用いて数量化し，連想の指標として用いる。連想検索の形態は，きっかけとなるタップルから他のタップルを導く形式である。連想検索の2，3の特性を示し，連想検索を繰り返した際の検索結果の振舞について考察を行う。

1．はじめに

1970年，E.F.Codd博士によって提唱されたデータのリレーショナルデータモデル[1]は，それまでのデータベースの概念を大きく変えた。リレーショナルモデルにもとづくリレーショナルデータベースは，高度のデータ独立性を実現すると伴に，利用者からはデータベースを表の集まりと単純に捉えるだけでよく，その表の集まりに対して施せる検索・更新等の操作が柔軟かつ強力である特長を持っている。現在のデータベースシステム（データベースとデータベースマネジメントシステムを合せデータベースシステムと呼ぶ）の主流は，CODASYL型データベースシステムからリレーショナルデータベースシステムに移った。

一方，電子計算機技術の進展は，大型計算機からパーソナルコンピュータまでの種々のクラスのデータベースシステムの稼働を可能とした。データベースシステムの利用者は，数年前まで巨大なコンピュータシステムで稼働していたデータベースシステムを，今や個人で簡単に利用でき，データベースの構築も容易にできるようになった。利用者の拡大のにより，より柔軟な仕様のシステムが求められている。

データベースシステムの柔軟な仕様の一つとして，連想検索が考えられる。本稿では，リレーショナルデータベースシステムにおける連想検索機能を提案する。提案する連想検索機能は，研究支援などを目的とした個人あるいは少人数の利用を前提としたものである。

この前提は，データベースの規模を指すものではなく，データが利用者の一定の意思にもとづく基準により数量(計量)化等がなされたもので，その基準に重要な意味をもつようなデータで構成されたデータベースを指すものである。すなわち，データベースが汎用的なものでなく一定目的のために構築されたものであることを意味する。

本稿で述べる連想検索機能は，リレーショナルデータベースを一枚の表(第1正規型)[1)]と捉えたとき，行(組，タップル)から行(組，タップル)を検索する演算(連想検索演算)を定義する。連想の指標をして，エントロピーを用いたタップルの評価を導入する。また，タップルからタップルへ次々と連想検索を繰り返した場合について議論する。以下，リレーショナルデータベースシステムについて簡単に述べ，エントロピーを用いた成分およびタップルの評価方法を示す。次に，連想検索機能(直接連想，高次連想)について定義し，連想結果について考察を行う。

2．リレーショナルデータモデル

2．1　データモデル

データベースシステムの構築は，データベース化したい対象世界を認識することから始まる。しかし，対象世界をそのままコンピュータ内に取り込むことはできない。そこで，物理的，論理的に許される範囲でデータ構造のモデル化が考えられる。このモデル化により示されるモデルは，データモデルと呼ばれている。

データモデルは，データベースおよびデータベースマネジメントシステムを設計・構築する上で重要な位置を占める。データモデルは，3つの階層(三層スキーマ構造：外部，概念，内部)[2)]から普通構成される(図1参照)。外部スキーマは，エンドユーザ，応用プログラム側から観たモデルである。概念スキーマは，対象世界の情報構造を計算機の論理データ構造に移したモデルである。内部スキーマは，概念スキーマを実現する物理的構造のモデルをいう。本研究は，連想を概念スキーマの立場でモデル化を行うことを目標としている。次に，リレーショナルデータモデルを簡単に示す。データモデルについての一般的議論は，本稿の主旨から外れるのでこれ以上行わない[1, 3)]。

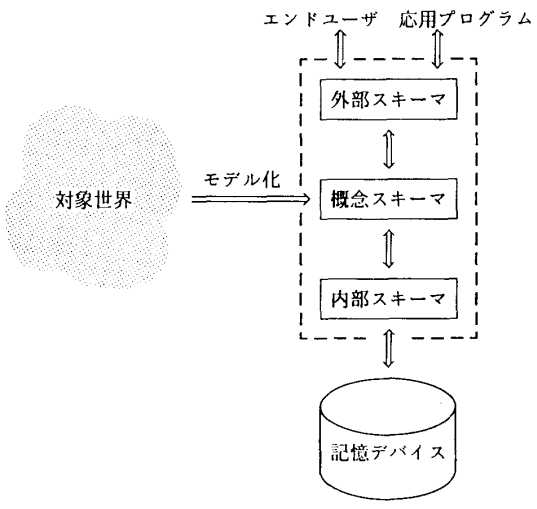

図1　データモデルの三層スキーマ構造

⇕：インタフェース

2．2　リレーショナルデータモデル

リレーショナルデータモデルは，数学の集合と関係の概念にもとづき対象世界の情報構造をモデル化したものである。以下，リレーショナルデータモデルの構成を簡単に述べる。

集合列$D_1,D_2,\cdots,D_n$が与えられるものと仮定する。これらの集合の直積$\Pi_{i=1}^{n}D_i$の部分集合を関係と呼び，関係を適当に集めてできる集合族Rを関係の型という。このときのnを関係あるいは関係の型の次数と言い，各D_iを定義域という。$D_1,D_2,\cdots,D_n$の中に同じ定義域が出現しても構わない。そこで，これらを区別するために改めて定義域に$A_1,A_2,\cdots,A_n$という互いに異なる名前を与える。この名前を属性名あるいは単に属性と言う。属性名は，集合全体(定義域全域)と同等に扱われる。

一つの関係の型は，属性を併記して

$$R[A_1,A_2,\cdots,A_n] \tag{1}$$

のように表わし，関係スキーマと呼ぶ。関係としては有限集合のみを考え，内包的記述法を用いた定義は

$$\begin{aligned} R=\{t \mid\ & t_1\in A_1\wedge t_2\in A_2\wedge\cdots\wedge t_n\in A_n\wedge \\ & t=<t_1,t_2,\cdots,t_n>\wedge r(t_1,t_2,\cdots,t_n)\} \end{aligned} \tag{2}$$

と表わされる。ここで$t=<t_1,t_2,\cdots,t_n>$は組(タップル)と呼ばれ，$r(t_1,t_2,\cdots,t_n)$はn項の述語で真か偽かの値を持つ。タップルを構成する個々の属性の値t_iを成分（属性値）という。

$R[A_1,A_2,\cdots,A_n]$において，属性の部分集合$\boldsymbol{A}=\{A_{k1},A_{k2},\cdots,A_{kn}\}$が，次の条件を満たすとき$A_{k1},A_{k2},\cdots,A_{kn}$を$R$のキー属性あるいは単にキーという。

(1) 関係R中に$A_{k1},A_{k2},\cdots,A_{kn}$の属性値を指定してやれば，$R$中のタップルは一意的に識別できる。

(2) $\boldsymbol{A}$は(1)の性質を満たす必要最小限の属性A_iからなる集合である。

キーは関係中に一種とは限らない。そこでキーのことを候補キーと呼び，特にどれか一つが主にキーの役割をもたせるとき，主キーと呼ぶ。

リレーショナルモデルにおけるデータベースは，時間とともに変化する関係の集まりと捉える。ある関係中のタップルと別の関係中のタップルは，キーを介して関連づけられ演算操作により目的の情報が選択される。関係に対する演算操作は，関係代数と呼ばれる代数系で与えられる。同時に一階述語論理にもとづく検索論理が，関係論理として提示されている。さらに関係論理で表現されるものは，すべて関係代数でも表現可能であること(関係完備性)が示されている[4]。

3．連想機能

データベースシステムに導入が期待される柔軟な機能として，連想機能が考えられる。データベースシステムにおける連想機能は，次のようにまとめることができる。

(1) データの蓄積段階における連想機能。

(2) データの検索段階における連想機能。

(3) データおよびシステム全体の管理に対する連想機能。

(1)は，データの整合性や不完全なデータの蓄積方法についての機能，(2)は，不完全な問い合せや曖昧な問い合せに対する機能，(3)は，(1)，(2)を総合的に矛盾なく管理する機構などと考える。この分類は大まかのものであり，さらに詳細な議論が必要と思われる。本稿では，(2)の検索段階における連想機能に焦点を絞り，以下論ずる。

3．1　連想検索

連想は，一般にある一つの概念から他の概念を連合して想起することである。また，概念は，複数の観念により形成されている。このように述べられる連想は，次のようにモデル化することができる。

概念は観念の集まりで，一つ概念は，一定の観念の組合せで決る。連想は，ある概念から他の概念への写像であると考えることができる。

リレーショナルデータベースシステムにおける連想検索を考えるに当り，まず，リレーショナルデータモデルの何が，概念や観念あるいは写像に当るかを考察する。

リレーショナルデータモデルでは，表は一つの関係，列は属性，行(タップル)は関係内での一つの事象を表わす。また，リレーショナルデータベースシステムにおける検索に必要な要求事項は，2次元の表から特定の列(属性)と成分(属性値)の範囲である。タップルは，複数の成分よりなることから，タップルに概念，成分に観念を対応させ，検索は写像に当ると考察できる。

以上の考察をもとにリレーショナルデータベースシステムにおけるタップル間連想検索を定義する。

タップル間連想は，あるきっかけとなるタップル(キータップル)t_kから他のタップルtへの写像

$$\alpha : t_k \longmapsto t \qquad t \in R \tag{3}$$

と定義し，この写像を連想写像と呼ぶ。連想写像の定義域は，関係Rと同じ次数のタップルであれば，Rに属している必要はない。値域は関係R全体である。

次に写像αを規定する必要がある。連想の意味から考え，キータップルに対し，なんらかの意味で近いタップルへの写像と考えるのが自然と思われる。そこで，タップル間の近さを表わす量としてタップル間類似度を導入する。以下，基本となる成分特徴量を定義

し，次にタップル間類似度を定義する。

なお，実際のリレーショナルデータベースは，複数の表により構成されているのが普通である。しかし，結合処理(演算)[1,4]を施すことにより1枚の表(第1正規型)に変換することが可能である。以下の議論では，第一正規型を前提として行う。

3．2　成分特徴量

関係はタップルで構成され，タップルは成分により構成されている。このことより，成分特徴量は，関係および各属性の秩序性を反映したものでなければならない。関係内で各属性ごとに成分を階層化し，成分を統計的に捉えることができるものとする。すなわち，属性(表の列)ごとに完全事象系をして扱えるものとする。

このときのあるタップルt_iのj番目の成分t_{ij}の成分特徴量 $s_{t_{ij}}$ を次のように定義する。

$$s_{t_{ij}}=p(t_{ij})\left(1-\frac{H_j}{H_{j\max}}\right) \tag{4}$$

ここで$p(t_{ij})$は，t_{ij}をj番目の成分としてもつタップルの(A_jの階層分けにしたがった)R全体に対する出現率であり，t_{ij}の一般性を表わしている。H_jは属性A_jのエントロピー[5]，$H_{j\max}$は属性A_jの最大エントロピー[5]を表わす。$\#R$を関係Rの濃度，$\#A_j$を属性A_jの濃度（階層数）とするとH_j及び$H_{j\max}$は，

$$H_j=-\sum_{h=1}^{\#R}p(t_{hj})\log_2 p(t_{hj}) \tag{5}$$

$$H_{j\max}=\log_2(\#A_j) \tag{6}$$

と示せ，$(1-H_j/H_{j\max})$は，属性A_jの秩序性を表わす量となっている。よって，成分特徴量$s_{t_{ij}}$は，t_{ij}の一般性を表わす量と属性A_jの秩序性を表わす量との積で表わされる量となっている。$s_{t_{ij}}$は，0から1まで値を採り，大きいほどt_{ij}が特徴的であることを表わす。

3．3　タップル間類似度

タップル間類似度は，2つのタップルの近さ，遠さを表わす基準として導入する。二つのタップルt_iとt_hの関係R上でのタップル間類似度$S(t_i,t_h)$は，成分特徴量$s_{t_{ij}}$を用いて，

$$S(t_i,t_h)=\frac{1}{n}\sum_{j=1}^{n}s_{t_{ij}}\delta_{t_{ij}t_{hj}}$$

$$\delta_{t_{ij}t_{hj}}=\begin{cases}1 & t_{ij}=t_{hj}\\ 0 & t_{ij}\neq t_{hj}\end{cases}：\text{Kronecher's delta} \tag{7}$$

と定義する。ここで，nは関係Rの次数である。

タップル間類似度$S(t_i,t_h)$は，二つのタップルt_iとt_hの各成分の一致，不一致を採り，一致した成分に対し，その成分の成分特徴量を掛合わせた値の平均である。

タップル間類似度は，対称律

$$S(t_i,t_h)=S(t_h,t_i) \tag{8}$$

を満たし，

$$0\leqq S(t_i,t_h)\leqq S(t_i,t_i)\leqq 1 \tag{9}$$

の値を採り，大きいほどt_iとt_hとが特徴的な点で近いことを表わしている。

4．タップル間連想検索

タップル間の連想は，3．1節で述べたようにキータップルt_kから他のタップルtへの連想写像αとして捉える。連想検索の結果はキータップルに対し，近いタップルが導かれることが期待される。タップル間の近い，遠いの尺度は，3．3節で定義したタップル間類似度を用いる。以下，2種類のタップル間連想検索を定義し，計算機実験から示唆された事柄を述べる。なお，計算機実験は，一様乱数および正規乱数を用いて各属性の階層数25，属性数(関係の次数)21，関係の濃度1000で行った。

4．1　直接連想検索

直接連想検索は，キータップルt_kから関係R内の他のタップルt_iを直接に導く。t_kからt_iを直接連想検索する連想の尺度として，直接連想係数$\rho(t_k, t_i)$を次のように定義する。

$$\rho(t_k,t_i)=\frac{S(t_k,t_i)}{S(t_k,t_k)} \tag{10}$$

この係数は，タップル間類似度をキータップルによる差異(特異性)を消去し正規化したものである。なお，キータップルt_kは，必ずしも関係Rに含まれている必要はない(図2参照)。

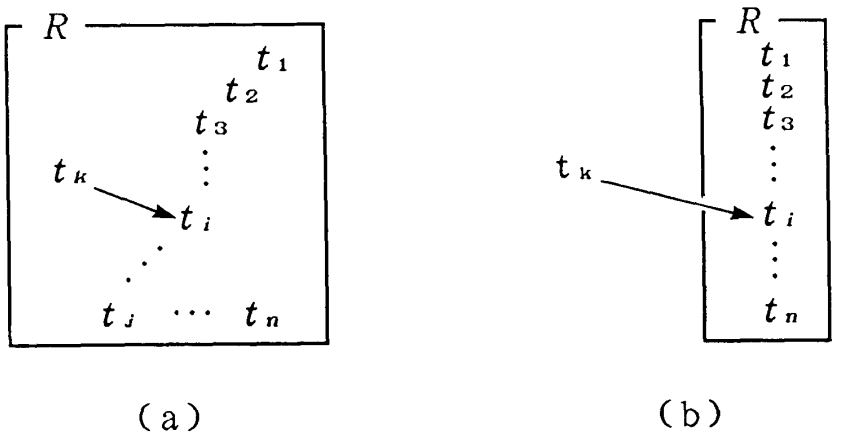

図2　直接連想検索

(a) キータップルが関係Rに属する場合。

(b) キータップルが関係Rに属さない場合。

$\rho(t_k, t_i)$は，連想のし易さ，し難さを表わすものであり，検索時のしきい値として使用する。すなわち，t_kと適当な直接連想係数値を与えることで，R内から複数のタップル検索することができる。また，直接連想係数が最大になるように検索を実行することも考えられる。このときの直接連想検索を単純直接連想検索と名づけ，直接連想係数を単純直接連想係数と呼び，

$$\rho(t_k,t_i)^* = \max_{i \in R}(t_k,t_i) \tag{11}$$

で示される。

直接連想係数$\rho(t_k, t_i)$の性質を次にまとめる。証明は，式(4)，(7)，(10)から容易にできるので省略する。

(1) $\rho(t_k,t_i) \in [0, 1]$

(2) $\rho(t_k,t_i) = 1$のとき必ずしも$t_k = t_i$とはならない。

(3) $S(t_k,t_i) = S(t_i,t_k)$であっても必ずしも$\rho(t_k,t_i) = \rho(t_i,t_k)$ではない。

(4) $\rho(t_k,t_i) = \rho(t_i,t_k) \Rightarrow p(t_{kh}) = p(t_{ih}); h = 1, 2, \cdots n$

4．2 高次連想検索

高次連想検索は，直接連想検索を逐次的に繰り返しタップルを導く。t_kをキータップルとする直接連想検索の結果は，しきい値に用いる直接連想係数の値により1個または複数個のタップルが検索される。高次連想検索の過程は，直接連想検索の結果，得られる1個または複数個のタップルの中から一つを選び，新たなキータップルとして，直接連想検索を繰り返す形式を採る。

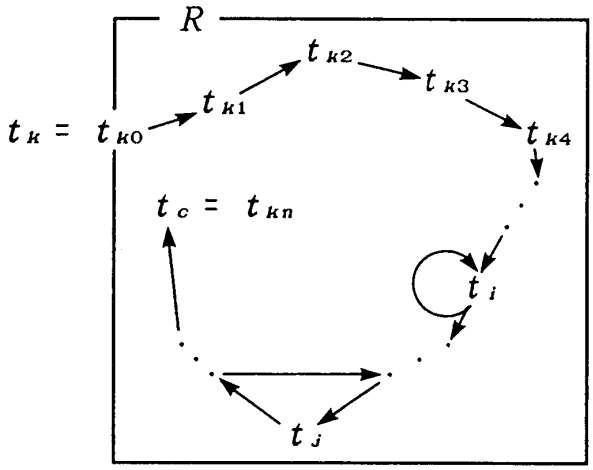

図3　高次連想検索

t_kをキータップルとして高次連想検索を行い，タップル列Π_{kc}を経由してタップルt_cを得たとき，タップル列Π_{kc}に沿った高次連想と言う（図3参照）。タップル列は，t_kから逐次的に直接連想検索を行った結果をt_{k1}，t_{k2}，…，t_{kn-1}，t_cと並べ，

$$\Pi_{kc}=t_{k0}t_{k1}t_{k2}\cdots t_{kc-1}t_{kn} \qquad t_{k0}=t_k\,;\,t_{kn}=t_c \tag{12}$$

と示す。タップル列Π_{kc}に沿った高次連想検索の難易を表わす尺度として，高次連想係数$\mu(t_k,t_c)/\Pi_{kc}$を直接連想係数をもとに，次のように定義する。

$$\mu(t_k,t_c)/\Pi_{kc}=\min_{0\le i\le n-1}\rho(t_{ki},t_{ki+1}) \qquad t_{k0}=t_k\,;\,t_{kn}=t_0 \tag{13}$$

この係数は，t_kからt_cへ至るタップル列の隣接するタップル間の直接連想係数の最も小さい値である。すなわち，タップル列Π_{kc}に沿った高次連想係数は，タップル列Π_{kc}中で最も連想し難い直接連想係数をもって定まる。

t_kからt_cへ至るタップル列は，しきい値および直接連想検索の結果の複数のタップルからいずれを採択するかにより複数存在する。したがって，高次連想係数も一意には定まらない。そこで，タップル列を単純直連想検索のみを逐次的に繰り返す高次連想検索の場合を考える。このとき得られるタップル列を単純タップル列Π_{kc}^*と示す。また，Π_{kc}^*に沿った高次連想検索およびその高次連想係数を，単純高次連想検索および単純高次連想係数$\mu(t_k,t_c)/\Pi_{Kc}^*$と名づける。単純高次連想係数は，一意的に定まり，タップル列は，同じ単純直接連想係数もつものが存在しない限り一意に決定される。

4．3　単純タップル列Π_{kc}*についての考察

単純タップル列Π_{kc}*は,単純直接連想検索の結果にしたがって決定される。このことは,R内の任意の二つのタップル間に必ず単純タップル列が存在するとは限らないことを意味する。また,Rが有限集合であることから,単純高次連想検索を無限に続けたとき,Π_{kc}*は長さlの“しっぽ”を前にもった周期mのループの構造[6]となる。すなわち,Π_{kc}のi番目のタップルt_{ki}を$t_k\tau_{m,l(i)}$と示すと

$$\tau_{m,l(i)}=\begin{cases} i & i<l \\ l+(i \mod m) & i\geqq l \end{cases} \quad n=1,2,3,\cdots \qquad (14)$$

のようになる。ここで（・mod m）は，mを法とする数を表わす。有向グラフで表わすと図4のように示せる。ループを形成する部分の隣接したタップル間では，単純直接連想係数が等しくなっている。

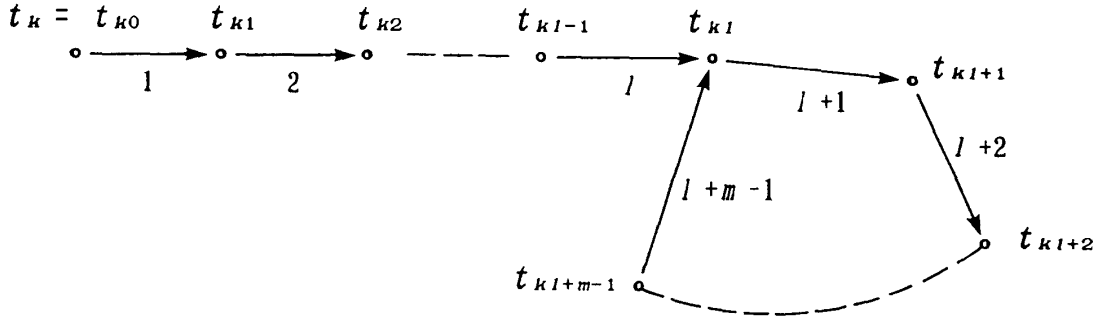

図4　長さlの“しっぽ”をもった周期mのループの構造

Π_{kc}*構造は，連想の遷移過程を示すものである。Π_{kc}*は，式(11)により定めるが，実際はさらに加えて，R内での検索順序と連想検索の停止条件により決定される。そこで，検索順序が一定で同じ単純直接連想係数を持つタップルが複数個存在する場合，最初に検索されたものを採択することにし，停止条件としてすでにタップル列に採択したタップルが再び採択されたとき検索を停止するものとする。そのとき，単純高次連想検索は，長さlの“しっぽ”を前にもった周期2のループの構造の単純タップル列Π_{kc}*を形成した時点で停止する。

ループが周期2になる理由は，Π_{ki}*の互に隣りあうタップル間の類似度に対して，推移律(単調増加)を満たすことからループとなるとき，検索順序の条件より必ず一つ手前のタップルに戻ることになるためである。

周期2のループの両端になるタップルを吸収タップル対と名づける。一つの関係内の吸収タップル対は，必ず存在する。この対のタップル相互の類似度は一致するが，単純直接連想係数は，一般には一致しない。吸収タップル対に付随する2つの単純直接連想係数のうち大きい側を主吸収連想係数と名づけ，その直接連想を主吸収連想，キータップルとなる側のタップルを主吸収タップルと名づける。一般的に一つの関係内の吸収タップル対は，複数個存在すると考えられる。

5．おわりに

本稿では，リレーショナルデータベースシステムにおける柔軟な機能としての連想検索機能を提案した。データベースに蓄えられているデータを階層化し，データベース内を完全事象系とする仮定のもとで，エントロピーを用いて成分特徴量とタップル間類似度を定義し，タップル間連想検索(直接連想検索および高次連想検索)を定式化した。また，連想係数，タップル列等の定義を行い，それらについての考察を述べた。

提案した連想検索機能は，データモデル(概念スキーマ)上でのモデルであり，実際計算機上にインプリメントするには多くの問題があると思われる。また，データを階層化により完全事象系として扱う仮定には，さらに十分な検討を要するものと考えられる。しかしながら，本稿の最初で述べたように個人仕様や研究支援目的のデータベースに対し，階層化そのものが大きな意味を持つ場合，ここに示した検索機能は，興味深い結果を導いてくれることが期待できる。さらに，単純連想検索や単純高次連想検索が，局所的ではあるが人間の連想に近い振舞(吸収タップル対の存在→発想の迷路)の簡単なモデルになっているものと考えられる。

データベースシステムにおける連想機能をはじめとする柔軟な機能は，第3章で述べたように検索以外にも多くの点で，今後必要不可欠なものと思われる。本稿で示した連想検索機能が直接役立つものとは思われないが，データベース全体が反映された量を重みに検索の評価を行う着想は，従来，行われてきた検索と異なった形式であり，柔軟性を持った処理を生み出すことが期待できる。今後の課題としては，現実のシステムおよびデータを用いた連想検索実験を行うことがまず挙げられる。実データを用いることで連想検索機能に留らず，真の柔軟な処理が明らかになってくるものと考えられる。

なお，本研究の一部は，文部省科学研究費補助金(奨励研究(A)特別研究員No.630790435)によった。

参考文献

1) E. F. Codd : A Relational Model of Data for Large Shared Data Banks, Comm., ACM, Vol. 13, No. 6, pp377-387(1970).

2) ANIS／X3／SPARC Study Group on Data Base Management System, Interim Report, FDT-Bulletin of ACM SIGMOD, Vol. 7, No. 2(1975).

3) 穂鷹 良介：データベースの論理設計(1), 情報処理, Vol. 24, No. 5, pp. 651-665(1983).

4) E. F. Codd : Relational Completeness of Data Base Sublanguages, in Courant Computer Scienc Symposium 6, Data Base System, pp. 65-98, Prentice-Hall, New Jersey(1972).

5) 小沢 一雅：情報理論の基礎, p.165, 国民科学社, 東京 (1980).

6) A. Gill : Semigroups and Monoids, Applied Algebra for the Computer Sciences, pp. 266-270, Prentice-Hall, New Jersey(1976).

Associative Retrieval Modelling for Relational Database Systems

Tunekazu KATO*, Kazumasa OZAWA** and Kuninosuke IMAEDA***

*Okayama University of Science and
JSPS Fellowships for Japanese Junior Scientists,
1−1 Ridaicho, Okayama 700 Japan

**Department of Management Engineering, Faculty of Engineering,
Osaka Electro-Communication University,
18-8, Hatu-cho Neyagawa-shi, Osaka, 572, Japan.

***Department of Electronic Engineering, Faculty of Engineering,
Okayama University of Science,
1−1 Ridaicho, Okayama 700 Japan

(Received September 30, 1989)

A database system is required to have more flexible retrieval function. In this paper, we propose an associative retrieval model for the flexibility on a relational database system. The model is of a retrieval type that retrieves from one key tuple to other tuples in a database. We show two types of retrieval models : one is the direct association and another the successive association. By using the "entropy" concept, the characteristics of the data is quantified. Then the degree of association is also quantified. The model is discussed from the view point of the mathematical properties of the degree of association.