

3H-8

出力生成の最適化を行う
非正規出力インタフェースの実現法

山本 秀樹 上林 弥彦
(九州大学 工学部)

1. まえがき

関係データベースシステムの普及に伴い、利用者が直観的に理解しやすい出力が求められるようになった。直観的に理解しやすい関係として、非正規形が知られている。そこで、関係データベースシステムの第一正規形出力を、非正規形に変換するインタフェースを通すことにより、非正規形による出力を実現する方法について考察する。このインタフェースを非正規出力インタフェースと呼ぶ。非正規化効率のよいインタフェースを実現するために、正規形の冗長度が下げるように質問を分解する。本稿では、このように質問を分解することにより非正規形生成効率を改善する、非正規出力インタフェースの実現法について述べる。

2. 基本的事項

データベースに関する基本的な事項は[4]に準ずる。各属性の定義域が単純値、集合値あるいはある属性集合上の関係である関係を非正規形と言う。正規形から非正規形を生成するための操作として、Group-byとRow-nestを用いる。[1]

Group-by[X]は、属性集合Xの値の同じ部分に注目して関係の横切りを行い、その後Xの部分について同じ部分関係内の組をまとめる操作である。関係の横切りとは、一つの関係に含まれる組を分割し、それぞれの組集合をまとめて表示する操作である。Row-nest[X]は、X以外の属性に対しGroup-by操作を施した後、一つの横切り部分に対応するXの値を一つの集合要素としてまとめる操作である。Group-byは関数従属性、Row-nestは結合従属性を表現するのに適している。

本稿では、Row-nestは非正規関係の結合で扱うので、非正規化操作としてGroup-by操作だけを考える。また、非正規化は一つの属性で行い、データベースシステムからの出力は非巡回結合従属性を満足していると仮定する。関係内では集合を表わすために[]を用いる。図1にRow-nest[趣味], Row-nest[成績, 科目], Group-by[成績]の順に非正規化を行った結果できた非正規形を示す。非正規形は親子関係を表わしているとみなすことができる。図1の関係全体では、「学生名」が親、「趣味」, {成績, 科目}が

学生名	趣味	成績	科目
前田	野球 ドライブ	優	英語 数学
		可	物理
西村	映画	良	地理 生物

図1 非正規化操作

子であり、{成績, 科目}では「成績」が親、「科目」が子になる。

3. 非正規出力インタフェースの構成

非正規出力を関係データベースシステムの正規形出力を非正規化することにより実現する。しかしデータベースシステムの出力を一つの正規形で表現すると関係の冗長度が高くなる場合があるので、出力を複数の正規形に分解する。出力を分解するために、利用者の質問を分解して実行する。図2に本稿で扱う利用者の質問を分解する非正規出力インタフェースを示す。この方法を用いた時に考察すべき事柄は次の三点である。

- (a) 質問をどのように分解すれば非正規化効率を改善できるか。
- (b) 質問を実行した結果得られた正規形をどのように非正規化すればよいか。
- (c) 質問を実行した結果得られた分解した非正規形をどのように結合するか。

[3]では、(c)の結合処理のための計算量を減少させるために、(a)で計算量が多くなる結合を避けるように質問を分解する方法を用いた。この方法の問題点は、データベースシステムからの正規形出力の冗長度が高くなり、正規形を記憶するために必要なメモリ量が多いことである。そこで、本稿で用いる方法では質問は出力が満足する結合従属性だけに基づいて分解する。この方法では、結合のための計算量が増加するが、正規形出力の冗長度は最も低くなり、正規形を記憶するのに必要なメモリ量は最少に

なる。

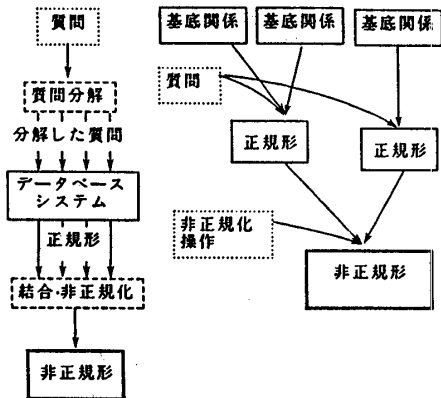


図2 質問分解する非正規出力インタフェース

4. 非正規形の生成

非正規形の生成を次のような順序で行う。

- I 質問に対する出力結果が満足する意味制約で、出力の冗長度が最も高くなるのは結合従属性である。そこで、質問と非正規化操作が与えられると、非正規出力インタフェースはデータベースシステム内の基底関係と質問の干渉によって生じる結合従属性を求める。このような結合従属性を求めるアルゴリズムは[1]に示されている。
- II 質問を結合従属性を構成する属性集合に対する部分質問に分解する。質問を分解することにより、データベースシステムから得られる出力結果の冗長度は低くなると考えられる。
- III IIの出力結果の非正規化を行う。非正規化操作は親子関係を作り出す操作と考えられるので、非正規化操作系列を、親子集合を表現するバックマン線図で表わす。さらに質問を分解する方法を用いた時、バックマン線図には次の二種類
 - (1) 出力する非正規形に対応するバックマン線図
 - (2) 分解した各関係に対応するバックマン線図を合成したもの

が考えられる。本稿では、(1)を非正規化グラフ、(2)を非正規化順序グラフと呼ぶ。非正規化グラフと出力が満足する結合従属性から非正規化順序グラフを作るには、非正規化グラフの閉包を求め、結合従属性の複数の構成要素におよぶ枝と、一つの構成要素内で他の枝から推移的に導くことができる枝を取り除けばよい。

出力結果の非正規化は、出力が満足する結合従属性上の非正規化順序グラフに従って行う。各出力結果について、その上で成り立つ非正規化順序グラフの根属性から始め、グラフの枝の順にGroup-byを行う。

- IV IIIの結果えられた分解した非正規形の結合を行い、不必要な属性を射影により取り除くことにより、求める非正規形を生成する。非正規形の結合は正規形の結合とは異なり、三通りの場合が考

えられる[2]。

図3の例で、非正規形生成処理を説明する。正規形出力が図3の(a)の結合従属性を満足している時、(b)の非正規化操作を施せば(c)のような非正規形がえられる。この非正規形の非正規化グラフは(d)であり、非正規化順序グラフは(e)である。

- I 利用者の質問と基底関係から(a)の結合従属性を求める。
- II 属性集合{A, B, C, D, E, F}についての質問を(a)の結合従属性に基づいて、属性集合{B, C}, {C, A}, {A, D}, {D, E}, {D, F}に対する五つの部分質問に分解する。分解した各部分質問を実行し、五つの正規形出力を得る。
- III IIで求めた正規形出力を(e)の非正規化順序グラフに基づいて非正規化する。例えば、関係R(B, C)ではグラフの向きはBからCの方向なので、BがCの親になるようにGroup-by[B]を行う。これを求めた全ての正規形関係に対して行う。
- IV IIIで生成した分解した非正規形を結合して、一つの非正規形を作る。一般には、結合後に不必要な結合属性を射影で取り除くが、この例ではそのような属性がないので、結合してできた非正規形が求める非正規形である。

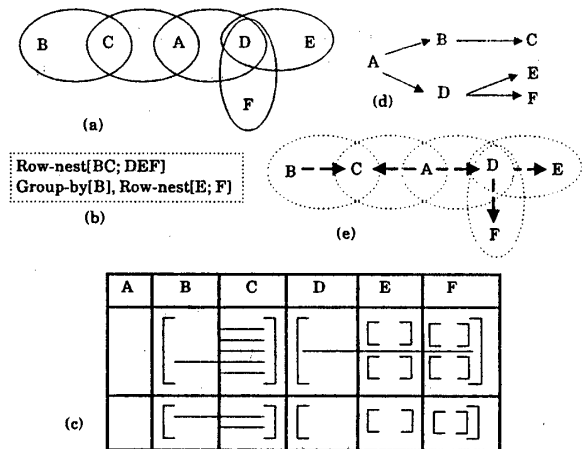


図3 質問分解を用いた非正規形生成例

参考文献

- [1] Kambayashi, Y. Tanaka, K. Takeda, K. "Synthesis of Unnormalized Relations Incorporating More Meaning," Information Sciences, Vol. 29, pp. 201-247, 1983.
- [2] 山本, 上林, "関係データベースの非正規出力のための最適化," 情報処理学会データベースシステム研究会 54-1, 1986.
- [3] 山本, 上林, "質問分解による会話型非正規出力インタフェース," 情報処理学会第32回(昭和61年前期)全国大会, pp.867-868, 1986.
- [4] Ullman, J.D., Principles of Database Systems, 2nd Edition, Computer Science Press, 1982.