情　報　学　基　礎　32－2
（1993．11．9）

関数従属性と包含従属性の相互作用をも対象とした場合の正規形データベーススキーム作成手法

山田　光博　　　　　　中川　優

NTT情報通信網研究所

従来，普遍関係の存在を仮定してその上の関数従属性を与えた場合に第三正規形データベーススキーム作成を行う手法が多数提案されてきた．しかし，普遍関係の存在と言う仮定は強く，実世界の概念を図式化する際に適したモデルであるERモデルおよびIFOモデル等で表現可能な情報の中には，普遍関係とその上の関数従属性制約では表現不可能なものが存在する．そこで本稿では意味表現力を高めるために複数の関係スキームを仮定し，各スキーム上の関数従属性制約およびスキーム間の包含従属性制約を与える方法を考える．更に，本条件の元で，各スキーム内の関数従属性に関して，関数従属性固有の導出律のみならず包含従属性との相互作用に関する代表的な導出規則であるPull back規則をも対象とした場合の導出手法およびそれを用いた正規形データベーススキーム作成手法を提案する．

提案手法の特徴は，最初に与えられるデータベーススキームを超グラフに類似のモデルで表現し，その上の変形により関数従属性の導出および正規形データベーススキーム作成を行う点である．

A Method for Synthesizing Normal Form Database Schemes Under Functional and Inclusion Dependencies

Mitsuhiro　YAMADA　　　　Masaru　NAKAGAWA

NTT Network Information Systems Laboratories

There are many synthesizing methods for normal form database schemes from the universe of attributes and the set of functional dependencies (FD) on it. But existence assumption of universal relation is too restrictive to allow the model represent the real world sufficiently. Therefore we adopt a method capturing real world with database scheme consist of relational schemes, FD sets on them and inclusion dependencies(IND)sets between them.

This paper proposes a derivation method to determine FD-covering on each relation scheme under not only FD's proper derivation axioms but also pullback rule that is one of representative derivation axiom of interaction between FD and IND sets. This method transform database schemes initially given to normal form database schemes. The characters of proposed methods are that it represent database scheme initially given with the model similar to Hypergraph and derivation of FD and that normalization of database scheme is executed by transformations on this model.

1．まえがき

属性の全体集合上の普遍関係の存在を仮定してその上の関数従属性（FD）制約を与えた場合，これから第三正規形データベーススキーム作成を行うための手続きとして，Berri and Bernsteinの設計法[1]・Bernsteinの合成設計法[2]に代表される種々の方法が提案されて来た[1,2,3,4,5]．しかし，互いに関数従属する（同値な（equivalent[1,2]））属性が存在する場合には，文献(1)および(2)の方法に基づく第三正規化手続きでは，同値な属性（候補キー）をマージする前後で，冗長な関数従属性（redundant FD[1,2]）の排除を二度行う必要が生じる．例えば，関数従属性集合{AB→C, AB→E,DE→A, DE→B, ACG→D,DE→F, F→C, F→H, C→G, H→C}に対しては，冗長な関数従属性の排除が繰り返されることになる．（詳細は章2 参照）．

一方，普遍関係上に関数従属性制約を記述するだけでなく，実世界の概念を図式化するのに適したより意味表現力の高いモデル[6,7,8]をも補助的に用いる方法も議論されている[6,7]．この場合，その設計結果を普遍関係上で表現することが困難な場合がある[9]ため，複数の関係を仮定し，関係スキーム間制約として包含従属性を与える方法が議論されてきた[7,10,11]．更に，関数従属性と包含従属性が混在する場合には，従来の普遍関係上の正規化手法では主キーに推移従属する属性を含む関係スキーム作成を行う場合がある．例えばデータベーススキーム（DBS）（＜R1(ABC), A→C＞，＜R2(AB), A→B＞，＜R3(BC), B→C＞, {R1(BC)⊆R3(BC)}）が与えられる（詳細は章２参照）．

そこで本稿では，冗長な属性（extraneous attributes[1][2]）を，同値なキーをマージする際に生じるもの（導出型の冗長な属性）と，そうでないもの（非導出型の冗長な属性）に分類した．そして，非導出型の属性を排除した後，同値な属性をマージし，その後非導出型の冗長な属性と冗長なFDの排除を同時に行うことにより，冗長なFDおよび属性の排除を繰り返さない手法を提案する．ここで，同値な候補キーおよびそのマージの特徴付けが容易となるように，超グラフに類似のモデルを用いて関数従属性の集合を表現する．更に，第三正規形関係スキーム作成に必要な各過程に対応するモデル上の変形操作を定義する．本稿の手法の特徴はこれらの各変形操作をモデルに一度限り適用することで実現される．

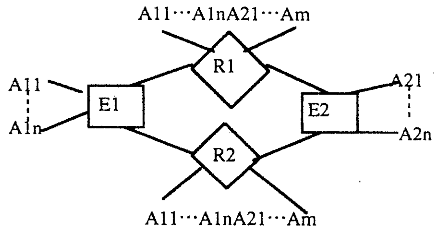
また，従来の第三正規化手法では主キーに推移従属する属性を排除不可能な場合があるという問題に対しては，関数従属性の導出に関して，それ固有の導出律のみではなく包含従属性との相互作用に対する健全な（sound）導出律の一つであるPull back規則[12,13]をも対象とした第三正規形（IND第三正規形）関係スキーム作成を行う方法が提案されている[14]．本稿では，IND第三正規形をより形式的に定義し，超グラフに類似のモデル上の変形によるIND第三正規形データベーススキーム作成手法を提案する．本手法ではデータベーススキームd=（＜R1, F1＞，…，＜Rn, Fn＞，I）を，IND第三正規形データベーススキームd'=（＜R1', F'1＞，…，＜R'n, F'n＞，I）に変形するが，特にdが普遍関係とその上の関数従属性集合である場合には従来の第三正規形データベーススキーム作成を行う．しかも，上述したように各関係スキームの第三正規化を行う過程で冗長な関数従属性の排除を同値な属性のマージの前後で繰り返さない．

2．IND第三正規形データベーススキーム

2-1　対象とするモデル

従来，関係スキーム設計を行う場合に，属性全てを含む普遍関係を仮定し，その上の関数従属性制約を与えることによって，管理対象およびそれらで管理する属性項目間の関係を捉えて来た．ところが普遍関係が存在するという仮定は非常に強い制約であり，これによって表現不可能な対象およびその管理属性間の関数従属性が存在する．

例えば，実世界の概念を図式化する段階においては，IFOモデルおよびERモデル等のより意味表現力の高いモデルが用いられるが，これらで表現可能な情報の中には普遍関係上でFD制約を与えるのでは表現不可能なものがある．例えば，同一の二実体間に異る関連が存在する場合には，それらを普遍関係上の関数従属性集合により表現することは不可能である（図１）．



制約：E1(A1i)=R1(A1i), R1(A2i)=E2(A2i)
E1(A1i)=R2(A1i), R2(A2i)=E2(A2i)

A11…A1n	A21…A2m	
(a1)11…(a1)1n	(a1)21…(a1)2m	R1
…………		
(ak)11…(ak)1n	(ak)21…(ak)2m	
(a1)11…(a1)1n	(b1)21…(b1)2m	R2
…………		
(ak)11…(ak)1n	(bk)21…(bk)2m	

∃s，tに対して，(bs)2t≠(as)2t
∃tに対して，FD制約A11…A1n→A2tは成立しない．

図１　普遍関係上のFD制約では表現出来ない例

そこで本稿では，初期的に複数の関係スキームを与え，各スキーム上での関数従属性制約およびスキーム間の包含従属性制約を与え，その上で関数従属性固有の導出律に加え，包含従属性との相互作用であるpull back規則を対象とした場合の関数従属性に関する導出法を提案する．

2-2　基本的事項

本稿で用いる基本的な概念，記法に関して概説する．U＝{A,B,C, ……}を属性の全体集合（但し｜U｜＜∞）とし，d＝（＜R1, F1＞，…，＜Rn, Fn＞，I）をU上のデータベーススキームとする．ここでRiは関係スキーム（Uの部分集合），FiはRi内の制約である関数従属性（FD）fi　の集合，Ｉは関係スキーム間の制約である包含従属性（定義は後述）の集合とする．ここでU上の普遍関係が仮定され，そのうえの制約としてFD集合Fのみが与えられる場合には，特にd＝（＜U, F＞，ϕ）と記述する．更にX, Y, Z等で属性集合あるいは属性を表すものとする．lhs（ｆｉ）およびrhs（ｆｉ）で各々ｆｉの左辺および右辺を表し，lhs（Ｆｉ）＝$\cup_{fi \in Fi}$ lhs（ｆｉ），rhs（Ｆｉ）＝$\cup_{fi \in Fi}$ rhs（ｆｉ）とする，lhs（Ｆｉ）およびrhs（Ｆｉ）をそれぞれ$(Fi)_L$および$(Fi)_R$とかくこともある．単一の

属性からなる集合 {A} をA（従ってA⊂Xと書いてA∈Xを意味することもある），属性集合等に関する和X∪YをXYと略記する．以降，ある関係スキーム（属性の全体集合の場合もある）上の関数従属性集合を単にFと表記することもある．また，F^+でFのArmstrongの導出律[18]に関する閉包を表し，$X_F^+ = \{Y \mid X \to Y \in F^+\}$ とする．

［Armstrongの導出則[18]］

反射律：$Y \subseteq X \Rightarrow X \to Y$

推移律：$X \to Y,\ Y \to Z \Rightarrow X \to Z$

合併律：$X \to Y,\ X \to Z \Rightarrow X \to YZ$

更に，Fの表現を簡略にするため，本稿ではFに対して以下の仮定を置く．

［関数従属性集合Fに関する仮定］

1．$\forall f \in F$ に対し，（rhs（f）の元数）＝1

2．$\forall f: X \to A \in F$ に対し，$X \cap A = \phi$

関数従属性に関する合併律により仮定1．をおいてもFに関する一般性は損なわれない，仮定2．によってFは自明な関数従属性を含まない．関数従属性の反射律により，仮定2．をおいてもFに関する一般性は失われない．また，関数従属性をある関係上の制約としておいているので，以下が成り立つ．

3．$f: X \to B,\ g: X' \to B' \in F$ に対し，$X = X',\ B = B' \Rightarrow f = g$

次に，関係スキーム間制約である包含従属性（IND）[12,13]を以下で与える．

［包含従属性］

関係スキームR1およびR2，属性の順列X（⊆R1），Y（⊆R2）に関して，以下の(*)が常に成立するとき包含従属性R1[X]⊆R2[Y]が成立するという．

r1[X]⊆r2[Y]．但しriはRi上の関係であり，ri[X]はそのX上への射影．………(*)．以降R1[X]⊇R2[X]，R1[X]⊆R2[X]ときR1[X]＝R2[X]とかく．

更に，包含従属性に関しては，以下の健全かつ完全な導出律が存在する[12,13]．これらに関するIの閉包を以降ではI^+で表す．

［包含従属性に関する導出律[10,11]］

1. 反射律　R[X]⊇S[X]（X,Y等で属性のならびを表す）

2. 推移律　R[X]⊇S[Y]，S[Y]⊇T[Z] ⇒ R[X]⊇T[Z]

3. 置換・射影　R[A1,…,An]⊇S［B1,…,Bn］ のとき R[Ai1,…,Aik]⊇S[Bi1,…,Bik]　（{1,…,n}⊇{i1,…,ik}）

本稿では，対象とする包含従属性の範囲を，同一ラベルの属性の順列間に定義されるもの（typed inclusion dependencies[12,13]）に限定する．則ち，R［A1…An］⊇S［B1…Bn］を考えるのはAi＝Biのときのみとする．またFDとINDの相互作用（interaction）に関しては以下のpull back規則を仮定する（ここではtyped INDの場合に限って述べる）．

［Pull back規則[12,13]］

Ri［XY］⊇Rj［XY］，$X \to Y \in Fi$，⇒　$X \to Y \in Fj$

データベーススキームd＝（＜R1，F1＞，…，＜Rn，Fn＞，I）に対して，F（=F1⊔…⊔Fn）のIとの相互作用も含めた閉包$\overline{F}$を$\overline{F1}$⊔…⊔$\overline{Fn}$で定義する．但し$\overline{Fi}$は，各FjにArmstrongの導出律を，IにINDの導出律を更にPull back規則を有限回適用して得られるRi上のFD集合とする，F1⊔…⊔Fnは，集合F1，…Fnに対する直和とする．F1⊔…⊔Fn上では，異る関係上の両辺のラベルが同一の関数従属性は異る元である．上記の定義を用いて，IND第三正規形データベーススキームを定義する．

［IND第三正規形データベーススキーム］

以下を満たすデータベーススキーム（＜R1，F1＞，…，＜Rn，Fn＞，I）はIND第三正規形であるという．則ち，$\forall$iに対しRiが(*)を満たす（IND第三正規形である）こととする．

(*)…$\exists$Xi（⊆Ri，Xi→Ri∈$\overline{Fi}$）に対して以下が成り立つ．

(i)　$\forall$A∈Ri－lhs（Fi）に対し，以下を満たすY（⊂Ri）は存在しない．Xi→Y，Y→A∈$\overline{Fi}$かつ，Y→Xi∈$\overline{Fi}$でない．

(ii)　X'⊂Xiで，X'→Ri∈$\overline{Fi}$を満たすものは存在しない．

また$\overline{Fi}$⊇Fi^+であるので，RiがIND第三正規形であるときCoddの意味の第三正規形でもある．またI＝ϕの場合には$\overline{Fi}$＝Fi^+であるため，Riが第三正規形であることとIND第三正規形であることは同値となる．逆に，各関係スキームがCoddの意味の第三正規形であるにも関わらずIND−第三正規形ではない例を図2に示す．

D=（<R1 (ABC),{A→B, A→C}>,
<R2 (ABDE),{B→E, A→E, B→D},
{R1（AB）⊇R2（AB）}

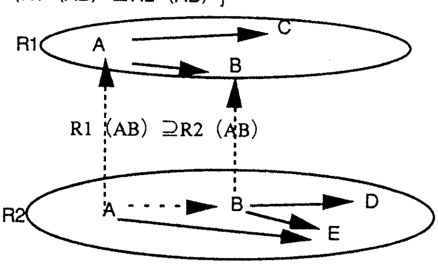


図2　各関係スキームが第三正規形であって
IND第三正規形ではない例

2-3　第三正規化手法のIND－第三正規化への拡張

Bernsteinの合成算法[2]が提案されて以来，データベーススキーム（＜U，F＞，ϕ）を第三正規形データベーススキーム（＜R1，F1＞，…，＜Rn，Fn＞，ϕ）に変形する手法が提案されてきた[1,2,3,4,5]が，本節ではこれをIND第三正規形の場合に拡張する．ここでは，データベーススキームd＝（＜R1，F1＞，…，＜Rn，Fn＞，I）でIが閉路を持たない場合に，Berri and Bernsteinの手法でF^+の部分を$\overline{Fi}$に置き換えた手法でdからIND第三正規形データベーススキームへの変形が可能となることを示す．最初に（<U,F>,ϕ）から第三正規形データベーススキームへの変形に必要な操作を示すために代表的手法であるBerri and Bernsteinの手法を示す

［Berri and Bernstein　の第三正規化手法[1]］

stepI-1　f：X→Bの左辺から冗長な（extoraneous[1]）属性Aを除く．（但し，Aが冗長⇔$X-A \to B \in F^+$）

stepI-2　Fから冗長な（redundant[1]）関数従属性を排除する．

（但し，fが冗長⇔f∈［F－f］$^+$　f：X→B）

stepII　同一の左辺を持つ関数従属性の，左辺および右辺の属性からなる関係スキームを作成し，共通の左辺をそのキーとする．

stepIII　同値なキー（equivalent keys[1]）を持つ関係スキームをマージする．

stepIV　IVにより生じる冗長な従属性および冗長な属性を排除する．

stepV　stepIVの終了結果に対し，stepIIを再度行う．

上記手法の各stepと同様の操作を各関係スキームRiに対して行うことにより，IND第三正規形データベーススキー

ム作成を行う方法を考える．そのためにまず，FDの左辺における冗長な属性および冗長なFDという概念を，FDの導出をFD集合内でのArmstrongの公理系によるものだけでなく，INDとの相互作用によるものをも対象とした場合に拡張する．

［F∪Ｉに関する冗長な属性］

データベーススキームd＝（＜R1，F1＞，…＜Rn，Fn＞，Ｉ），f ∈Fiに対し，lhs（f）－A→rhs（f）∈$\overline{Fi}$のときAをF∪Ｉに関する冗長な属性と呼ぶ．

［F∪Ｉに関する冗長なFD］

データベーススキームd＝（＜R1，F1＞，…＜Rn，Fn＞，Ｉ），f ∈Fiに対し，f∈$\overline{Fi-\{f\}}$のときf をF∪Ｉに関する冗長なFDと呼ぶ．

これらの概念を用いて，Berri and Bernsteinの第三正規化手法を以下の手法１に拡張する．

［手法１］各Riに対して以下を行う．

stepI-1　f：X→Bの左辺からF∪Ｉに関する冗長な属性を排除する．

stepI-2　FiからF∪Ｉに関する冗長なFDを排除する．

stepII　stepI-2終了時のFiにおいて，同一の左辺を持つFDの，左辺および右辺の属性からなる関係スキームRimを作成し，Fim＝｛key(Rim)→A|A∈Rim－key(Rim)｝とする

stepIII　同値なキー（key（Rim）←→key（Rin）∈$\overline{F∪I}$を持つ関係スキームRim，RinをRim∪Rinにマージし，その上のFD集合をFin∪Fimとする．stepIII終了後，RiがRi1，…Ripに分割されており，各々FD集合Fi1，…，Fipを持つとする．Fi=Fi1∪…∪Fipとする．

stepIV　IVにより生じるF∪Ｉに関して冗長な従属性および冗長な属性を排除する．

stepV　stepIVの終了結果に対し，stepIIを再度行う．

以下では，手法１がIND第三正規形データベーススキーム作成を行うことを示す．ここで$\overline{Fi}$の定義により，$\overline{Fi}$＝｛f｜lhs（f）∪rhs（f）⊆Ri，F∪I├…├f，但し，F∪I├…├fで├の個数回，Armstrongの導出律またはpull backを適用することにより，F∪Iからfが導かれることを表す｝．更に，$\underline{\mathit{Fi}}$＝｛f｜lhs（f）∪rhs（f）⊆Ri，F∪I├…├ f なる規則の適用├の列に対し，その最後の├がRi[X]⊆Rj[X]（∃X⊆Ri,∃j）およびFjの元に対するPull back規則の適用｝に対し，$\overline{Fi}$＝（Fi∪$\underline{\mathit{Fi}}$）$^+$である．ところが，Iに関して非閉路性が保証されている場合には，以下の命題に示す通り，$\underline{\mathit{Fi}}$はFj（j≠i）およびIのみに依存し，Fi自身には依存しない．

［命題１］d＝（＜R1，F1＞，…，＜Rn，Fn＞，I）において，R1[X1]⊆R2[X2]，…Rj[Xj]⊆R1[Xj+1]∈I'なるI'の元は存在しないものとする．このとき，∀g（∈Fi）に対し，

$\underline{\mathit{Fi-\{g\}}}$＝$\underline{\mathit{Fi}}$．

（証明）以下で定める有向グラフG=(V,E,L)を考える．

(1)　各関係スキームRiに対し，Vの元が唯一つ対応し，ラベルRiを持つ．

(2)　Ri[X]⊆Rj[X]∈I'に対し，有向辺（Rj,Ri）[X]∈E（ラベル X）が対応する．

(3)　各Riに対し，その上のループが唯一つ存在するものとする．

Iの非閉路性に関する仮定により，Gは有向閉路を持たない．Gの有効辺およびループに沿ったなぞりを以下の様に定義する．

(i)　Riのループを一回なぞることを，その時点で得られているRi上のFD集合Fi"にArmstrongの導出規則を一回適用し，新たに得られるFDをFi"に追加することに対応付ける．

(ii)　有向枝（Rj,Ri）[X]∈Eを一回なぞることを，Fj"の元とRi[X]⊆Rj[X]にpull back規則を適用し，その結果新たに得られるFDをFiに加えることに対応付ける．

この対応付けにより，∀f∈$\underline{\mathit{Fi}}$が与えられたときに，F∪I├…├ f なるArmstrongの導出律あるいはpull back規則の適用の系列に対し．Gの部分グラフG'(f)=（V'，E'，L）が一意に対応する．則ち，V'＝｛├がArmstrongの導出律の適用の場合に(i)により対応づけられる頂点｝∪｛├がpull back規則の場合に(ii)により対応付けられる有向枝の始点と終点｝，E'=｛├がpull back規則の場合に(ii)により対応付けられる有向枝｝である．このときG'(f)の各頂点からラベルRiの頂点（以降Riとかく）への有向道が存在する．また，Riは出次数=0である．実際，Riを始点とする有向枝があったとすると，その終点からRiへの有向道が存在し，G'(f)は有向閉路を持つことになる．ところがG'(f)⊆Gと，Gの非閉路性からこれは矛盾である．G'(f)の定義により出次数が0の点に対応する関係スキーム上のFD集合は$\underline{\mathit{Fi}}$の元の導出に無関係である．故に，f∈$\underline{\mathit{Fi-\{g\}}}$ □

Iに非閉路性を仮定する場合には，$\underline{\mathit{Fi}}$がFiに依存しない上記の事実を用いれば，手法１は各（＜Ri，Fi∪$\underline{\mathit{Fi}}$＞，ϕ）に対してBerri and Bernsteinの第三正規化手続きを適用することに帰着される．これにより以下が得られる（証明は付録参照）．

［命題2］データベーススキームd＝（＜R1，F1＞，…，＜Rn，Fn＞，I）において，手法１を適用することにより，以下を満たすIND第三正規形データベーススキームd'＝（＜R'11，F'11＞，…，＜R'1j1，F'1j1＞，…，＜R'n1，F'n1＞，…＜R'njn，F'njn＞，Ｉ）に変形される．

(i)　Ri＝R'i1∪…∪R'iji

(ii)　$\overline{Fi}$＝（F'i1∪…∪F'iji）$^+$

2-4　関数従属性（FD）に関して同値な属性

節2.2の手法１は各（＜Ri，Fi∪$\underline{\mathit{Fi}}$＞，ϕ）に（d=（＜U,F＞，ϕ）の場合には（＜U,F＞，ϕ）に）Berri and Bernsteinの第三正規化手法 [1] を適用する方法である．本節ではBerri and Bernsteinの手法の適用における同値なキーの存在の影響について述べる．本節では単にFと書いてFまたはFi∪$\underline{\mathit{Fi}}$を意味するものとする．同値なキーが存在する場合，stepI-1およびstepI-2で冗長な属性および冗長な関数従属性を排除したにも関わらず，stepIIIで同値なキーを持つ関係スキームをマージすることにより新たに生じる冗長な関数従属性および冗長な属性を，stepIVで再度排除する必要がある場合がある．例えば，F={AB→C, AB→E,DE→A, DE→B, ACG→D,DE→F, F→C, F→H, C→G, H→C｝の場合には，stepI-1およびstepI-2で冗長な属性および冗長な関数従属性を排除した後，stepIIIで同値なキーABとDEのマージを行うために生じる冗長な関数従属性AB（DE）→Cを再度排除する必要がある．

一方，冗長な属性に関しては，下記の命題１に示す通り二種類に分類される．

［命題３］　X'およびX（⊃X'）およびA（単一の属性）に対し，X→A∈Fが成り立つとする．このとき，X－X'→A∈F$^+$（X'⊂X）……①が成り立つには以下の(1)または(2)が成り立つことが必要十分である．

(1)　X－X'→X'∈［F－｛X→A｝］$^+$

（⇔　X－X'→X'，X－X'→X∈F$^+$）

(2)　X－X'→A∈［F－｛X→A｝］$^+$

（証明）　付録２参照

タイプ(2)の冗長な属性（以降非導出型の冗長な属性とよぶ）が存在する場合は，関数従属性X→B自身が冗長である．従って，非導出型の冗長な属性は冗長な関数従属性の排除に含めて行うことが可能である（以降，関数従属性X→Bの左辺から非導出型の冗長な属性Aを排除する操作も冗長な関数従属性X→Bの排除と呼ぶ）．また，下記の命題4が示す様に導出型の同値なキーをマージすることにより（下記命題4で，FからF'を作成する操作に対応する），導出型の冗長な属性が新たに発生することはない．以下に同値な候補キーのマージを具体的に記述し，命題4を示す．

［同値な候補キーをマージした関数従属性集合］

F＝{f1，…，fn} に対し，以下のF'を同値な候補キーをマージした関数従属性集合F'と呼ぶ．

F'＝（F－（F0∪F1∪F2∪F3））∪F4∪F5∪F6　但し，$F_{|I}$ は下記を満たすFの部分集合とする．

(i) $F_{|I}$ ＝ {fi1，…，fik｜lhs(fil)←→lhs(fim)∈F^+}

(ii) $^\forall$I'∈$2^{\{1,\cdots,n\}}$（I'⊃{i1，…，ik}），$^\exists$ip，iq∈Iに対lhs(fip)←→lhs(fiq)∈F^+ でない．

F0，F1，F2，F3，F4，F5は以下の通りとする．

F0＝$\bigcup_{|I|\geq 2}$ {lhs(f)→A（f∈$F_{|I}$）｜$^\exists$X∈lhs($F_{|I}$)，{A}⊆X}

F1＝$\bigcup_{|I|\geq 2}$ {lhs(f)→A（f∈$F_{|I}$）｜{A}∩lhs($F_{|I}$)＝ϕ}

F2＝$\bigcup_{|I|\geq 2}$ {X→A｜X∈lhs(F)－lhs($F_{|I}$)，A∈lhs($F_{|I}$)，|A|＝1}

F3＝$\bigcup_{|I|\geq 2}$ {X→A｜X∈lhs(F)－lhs($F_{|I}$)，A∈lhs(f)（f∈$F_{|I}$，|lhs(f)|　≧2）}

F4＝$\bigcup_{|I|\geq 2}$ {f''ij（ij∈I）|lhs(f''ij)＝lhs(fij)，lhs(fij)∈$F_{|I}$でなく，fij∈F2∪F3，rhs(f''ij)＝（lhs(fi1)）…（lhs(fik)），{i1，…，ik}＝I}

F5＝$\bigcup_{|I|\geq 2}$ {（lhs(fi1)）…（lhs(fik)）→A｜A∈（rhs($F_{|I}$)－lhs($F_{|I}$)）}

F6＝$\bigcup_{|I|\geq 2}$ {lhs(fij)→（lhs(fi1)）…（lhs(fik)）｜ij∈I，|lhs(fij)|≧2}

(lhs(fi1))…(lhs(fik))（I＝{i1,…,ik}）をcon（lhs（$F_{|I}$））と書いて縮退形と呼ぶことがある．

注1）定義より明らかに，fi（∈F）は唯一の$F_{|I}$ に含まれる．則ちF＝$F_{|I1}$∪…∪$F_{|In}$とすると，$F_{|Ii}$∩$F_{|Ij}$＝ϕ（i≠j）．

注2）|F4|≦|F2∪F3|，|F5|≦|F1|であるので，F＝$F_{|I1}$∪…∪$F_{|In}$に対し，|F'|≦|F|－|F0|＋|F6|＝|F|－|F0|＋$\Sigma_{|In|\geq 2}$ |m(In)|．但し，m(In)＝$F_{|In}$∩ {f||$F_{|I}$|≧2}＝{mk1,…,mki}．Len(f)≧2（f∈F），Len(f)≧3（f∈$\bigcup_{|I|\geq 2}$m(I)））|F'|≦|F－$\bigcup_{|Ik|\geq 2}$m(Ik)|＋|$\bigcup_{|Ik|\geq 2}$m(Ik)|＋|F6|≦1/2Len(F－$\bigcup_{|Ik|\geq 2}$m(Ik))＋$\Sigma_{|Ik|\geq 2}$|m(Ik)|＋$\Sigma_{|Ik|\geq 2}$|m(Ik)|≦1/2Len(F)-3/2$\Sigma_{|Ik|\geq 2}$|m(Ik)|＋2$\Sigma_{|Ik|\geq 2}$|m(Ik)|．また縮退型の属性を長さ1で表現すると，Len(F')≦Len(F－$\bigcup_{|Ik|\geq 2}$m(Ik))＋Len({con($F_{|Ik}$)→rhs(f)|f∈m(Ik)})＋Len(F6)≦Len(F)－3$\Sigma_{|Ik|\geq 2}$|m(Ik)|＋2$\Sigma_{|Ik|\geq 2}$|m(Ik)|＋2$\Sigma_{|Ik|\geq 2}$|m(Ik)|＝Len(F)＋$\Sigma_{|Ik|\geq 2}$|m(Ik)|．故にLen(f)≧3（f∈m(Ik)）からLen(F)≧3$\Sigma_{|Ik|\geq 2}$|m(Ik)|が導かれ，|F'|・Len(F')≦1/2Len$(F)^2$＋Len(F)・$\Sigma_{|Ik|\geq 2}$|m(Ik)|＋1/2（$\Sigma_{|Ik|\geq 2}$|m(Ik)|）^{2}≦Len$(F)^2$である．

注3）F1∪…∪F6は，Fの(*)の

［命題4］F'においてもFにおけるのと同様にArmstrongの反射律，推移律，合併律の成立を仮定する．またπ：lhs（F）→（lhs（F）－$\bigcup_{|I|\geq 2}$lhs（$F_{|I}$））∪con（$F_{|I}$）を，π（lhs(f)）＝lhs(f)（f∈F－$\bigcup_{|I|\geq 2}F_{|I}$ のとき），π（lhs(f)）＝con（$F_{|I}$）（$^\exists$Iに対し，f∈$F_{|I}$）で定義する．このとき，X→Y∈F^+（X，Y∈lhs（F）∪rhs（F）は以下の(1)または(2)が成立することと同値である．

(1)X，Y∈lhs（F）であり，$^\exists$Iに対し，π(X)＝π(Y)＝con（$F_{|I}$）

(2)π(X)≠π(Y)であり，π(X)→π(Y)∈$(F')^+$．

従って，特に次の(i)，(ii)が成立する．

(i)$^\forall$X∈lhs（F）に対して，X'⊂XかつX'→X∈F^+なるX'は存在しないことは，以下の①および②の成立と同値．

①$^\forall$Iに対して，con（$F_{|I}$）＝(X1)…(Xn)に対し，Xi⊇Xjとなる

i,jは存在しない．

②π(X)∈lhs（F'）－$\bigcup_{|I|\geq 2}$ con（$F_{|I}$）のときX'⊂Xかつ，X'→X∈$(F')^+$は存在しない．

(ii)π(X)→π(Y)∈［F－{π(X)→π(Y)}］$^+$（π(X)≠π(Y)）⇒X→Y∈［F－{X→Y}］$^+$．（証明）略．

更に，I に命題1の非閉路性を仮定している場合には，命題3は以下に拡張される（証明は付録参照）．

［命題5］データベーススキーム（d＝（<R1，F1>，…，<Rn，Fn>，I））が与えられ，Iは命題1の意味で非閉路性を満たし，X，X'，Aは命題3と同様であり，X→A∈Fiとする．このときX－X'→A∈$\overline{Fi}$……①が成り立つためには以下の(1)または(2)が成り立つことが必要十分である．

(1) X－X'→X'∈$\overline{Fi－\{X→A\}}$

（⇔ X－X'→X'，X－X'→X∈$\overline{Fi}$）

(2) X－X'→A∈$\overline{Fi－\{X→A\}}$

3.超グラフ的表現とその上の変形規則

3-1　データベーススキームの超グラフ表現

データベーススキームd＝（＜R1，F1＞，…，＜Rn，Fn＞，I）を超グラフに類似したモデルで表現する．本モデルFD-IND-HGは左辺が複数属性からなる関数従属性の表現等を容易にし，関係スキーム間の包含従属性を表現するために通常の超グラフ[17]とは異る形態を持つ．則ち，有向枝の始点となる超枝が存在する点，超枝間にそれに含まれる頂点と同一のラベルを持つ有向枝が存在するという特徴を持つ．前者は，複数属性からなる関数従属性の左辺を超枝で表現するためであり，後者は関係スキーム間の包含従属性を関係スキームに対応する超枝間の有向枝で表現するためである．

［FD-IND-HGとその構成要素］

与えられたデータベーススキームから，FD-IN-HG，H(F,I)＝（V,HE1,HE2,E1,E2）への対応を以下で定義する．

(i) $^\forall$A∈ {属性} に対し，ラベルAの頂点Va∈Vが# {Ri｜A∈Ri}個対応する．

(ii) $^\forall$lhs（f）（f∈F，｜lhs（f）｜≧2）に対し，ラベルlhs（f）の超枝he1（lhs（f））（$V_{lhs(f)}$と書くこともある）が#{i|f∈Fi} 個対応する．

(iii) 各Riに対し，ラベルRiのhe2(Ri)∈HE2が唯一つ対応する．$^\forall$A（∈Ri），$^\forall$f∈Fi に対し，Va，he1（lhs（f））∈he2（Ri）かつ# {Va（∈he2(Ri)）}＝# {he1（lhs（f））（∈he2(Ri)）}＝1．

(iv) ①$^\forall$ f：X→A∈Fiに対し，ラベル1の有向枝（Vx，Va）∈E1'（Vx，Va∈he2(Ri)）が唯一つ対応する．

②X＝A1…An（n≧2）なるfに対し，ラベル1'の有向枝（Vx，Vaj）∈E1''（Vx，Va∈he2(Ri)，j=1,…n）が対応する．

更に，E1＝E1'∪E1''とする．

(v) $^\forall$i（＝ Ri[X]⊆Rj[X]）∈I に対して，ラベルXの有向枝（he2[Rj]，he2[Ri]：X）（∈E2）が唯一つ対応する．

FD-IND-HG，H(F,I)の各構成要素の位数とデータベーススキームの構成要素の位数の間には以下の関係が成り立つ．

d＝（＜U，F＞，ϕ）の場合，｜HE1｜＋｜V｜≧p（但し，p＝｜F｜），｜HE1｜＝｜M｜（M⊂Fかつf∈Mに

対し｜les（f）｜≧2），特に｜HE1｜≦p，｜E1'｜=p，p≦｜E1｜≦Len（F），｜HE｜+｜V｜≦Len（F），但し，Len（F）は関数従属性集合Fの記述長である．例えば，F＝{ABC→D，CD→E，E→A}では，Len（F）＝9，p＝3，｜V｜=5，｜HE｜=2，｜E1｜=8である．またd=（<R1,F1>，…，<Rn,Fn>，I）の場合には，上で，pを|F1⊔…⊔Fn|で，Len（F）をLen（F1⊔…⊔Fn）でそれぞれ置き換えた関係が成り立つ．

3-2 FD導出のFD-ING-HG上での特徴づけ

IND第三正規形データベーススキーム作成に必要な関数従属性に関する導出規則による適用は，d=（<R1，F1>，…，<Rn，Fn>，I）において各Ri上で行われる．これは，FD-ING-HG，H(F,I)上で，その各he2∈HE2への制限上でFDの導出に対応する変形が行われることを意味する．ここで，FD-ING-HG，H(F,I)のhei∈HE2（i=1,…,n）への制限とはHi=（Vi，(HE1)i，(HE2)i，(E1)i，(E2)i）（ここで(Γ)i={γ∈Γ｜γ∈Γ}（Γ=V,HE1,HE2のとき），(Γ)i={γ∈Γ｜γ∈Γ}（Γ=E1,E2のとき））上でFDの導出規則に対応する以降これをFDHGi，Hiとよぶ．本節ではHiの変形規則，F^+にほぼ対応するHi上の概念としてFDHGiの閉包^{TR}を定義し，更に関数従属性に関して同値な候補キーの特徴づけを行うためにHi上に擬閉路を定義する．Hiの閉包は有向グラフの推移的閉包[20]，更にFD-graph[3]の閉包に類似の概念である．以降本節および次節ではFDHGi，HiをFDHG，H=（V,HE1,HE2,E1,E2）と略記する．

［FDHGの閉包］

以下の規則I，IIをFDHG，Hに，有限回適用することで得られ，以下を満たすものをFDHG，Hの推移的閉包を呼び，^{T}H=（V,HE1,HE2,^{T}E1,E2）と書く．

①^{T}E1（=^{T}（E1'∪E1"））はE1に以下の規則I，IIを有限回適　用することで得られる有向枝の集合．則ち，

^{T}E1=$\cup_{\gamma\in\Gamma}\gamma$（$T_I$，$T_{II}$）（E1）．但し，$T_k{}^m T_j{}^n$（E1）はFDHG　に規則j（j=I，II）をn　回適用したのち規則k（k=I，II）　をm回適用して得られる有向枝の集合であり，Γ=　{$x^n y^m$｜n,m∈N∪{0}}とする．

②T_I（^{T}E1）=E1　かつT_{II}（^{T}E1）=E1

注）^{T}E1は特に$\cup_j$（$T_I T_{II}$）^{j}（E1）で与えられ，従ってFDHGに対して一意に決る．

（閉包を生成するための規則）

規則I　Vx，Vy∈V∪HE1，Va∈Vに対し，（Vx，Vy）∈E1，（Vy，Va）∈E1'の場合には，（Vx，Va）をE1'に加える．（Vx，Va）のラベル付けは以下の通りに行う．

(1)　（Vx，Va）∈E1'（Vx，Va）のラベルαのとき）（Vx，Va）（∈E1'）のラベルにα+1"を加える．但し1"+1"=1"が成り立つものとする．

(2)　（Vx，Va）∈E1'ではないとき，（Vx，Va）をE1'に加え，そのラベルは1"とする．

規則II　Vx∈V∪HE1，Vy1…yn∈HE1（Vyi∈V）に対し，（Vx，Vyi）∈E1（i=1,…,n）かつ$^\exists$iに対して（Vx，Vyi）∈E1'のとき（Vx，Vy1…yn）（ラベル1"とする）をE1'に，（Vx，Vyi）∈E1"（i=1,…,n）のとき（Vx，Vy1…yn）（ラベル1"とする）をE1"に加える．

規則I　は，関数従属性の推移律を，規則II　は合併律をそれぞれFDHG，H上で表現するためのものである．

命題6に示す様に，Fを表現するFDHG，Hに対し^{T}Hを求めることにより関数従属性f（但し，lhs（f）∈lhs（F），rhs（f）∈rhs（F））がF^+に所属するか否かを判定出来る．

［命題6］X，Y∈F_L∪F_Rおよび．Vx，Vy∈V∪HE1（Vx，およびVyはラベルX，Yを持つとする．このとき，

X→Y∈F^+　⇔　（Vx，Vy）∈^{T}E1

（証明）　X(F,n)={X'∈F_L∪F_R｜Fにn回以内，Armstrongの導出律を適用することで，X→X'が得られる．}とし，Hに閉包を求めるための規則をn回以内適用して得られるE1'（E1"）の元の集合をE1'(n)（E1"(n)），E1(n)=E1'(n)∪E1"(n)とする．

（X→Y∈F^+⇒　（Vx，Vy）∈^{T}E1の証明）

(*)　Y∈X(F,n)　⇒　（Vx，Vy）∈E1(n)　を帰納法で示す．X(F,n)をXnで略記する．

n=0：X→Y∈Fより

(1)　X→Yが自明⇒（Vx，Vy）∈E1"（⊆^{T}E1）

(2)　X→Yが自明でない⇒（Vx，Vy）∈E1'（⊆^{T}E1）

nでの(*)の成立を仮定し，n+1での成立を以下で述べる．Y∈(Xn+1−Xn)のとき以下の(1)或は(2)のいづれかが成立する．

(1)　|Y|=1であり，$^\exists$Z∈Xnに対し，Z→Y∈F

(2)　|Y|≧2であり，Y1，Y2∈Xnに対し，Y=Y1Y2

(1)の場合：（Vx，Vz）∈E1(n)，(Vz，Vy）∈E1'．FDHG上の閉包を求めるための規則I　により，（Vx，Vz）∈E1'(n+1)

(2)の場合：最初に以下の(*)(*)を帰納法で示す．「Y=A1…Al∈Xn⇒（Vx，Va1），…，（Vx，Vah）∈E1(n)」…(*)(*).

初期段(n=0)：Fに関する「$^\forall$f（∈F）に対し|rhs（f）|=1」という仮定から，Y=$^\exists$A（∈rhs(F)）であり，（Vx，Va）∈E1(0).

帰納段（k≦nでの成立を仮定する）

Y=A1…Ah∈（Xn−Xn+1）（h≧2）とすると，Fに関する「$^\forall$f（∈F）に対し|rhs（f）|=1」という仮定から，$^\exists$s，t（≦n）に対し，Ai1…Aip∈Xs，Aj1…Ajq∈Xt（但しAi1…Aip∩Aj1…Ajq=φかつ，Ai1…Aip∪Aj1…Ajq=A1…Ah）．故に，帰納法の仮定により，（Vx，Vai1），…，（Vx，Vaip），　（Vx，Vaj1），…，（Vx，Vajq）∈E1(n).故にFDHGの閉包を求めるための規則IIを適用することにより，（Vx，Vy）∈E1(n+1).

(*)(*)により，Yi=Ai1…Aiji（i=1,2）について（Vx，Vai1），…，（Vx，Vaiji）∈E1(n).故に規則IIの適用により，Y=Y1Y2に対し（Vx，Vy）∈E1(n+1)を得る．

（（Vx，Vy）∈^{T}E1）　⇒　X→Y∈F^+の証明）

次の(1)および(2)が全ての自然数について成り立つことを示せば十分である．

①（Vx，Vy）∈E1(n)，（Vy，Va）∈E1'のとき，A∈Xn+1

②（Vx，V_{Ai}）∈E1(n)，Y={A1，…，An}のとき，Y∈Xn+1

帰納法で示す．

n=0：①の成立：（Vx，Vy）∈E1，（Vy，V_A）∈E1'．FDHGの定義により，X→Y∈F，Y→A∈F．故にArmstrongの推移律によりA∈X1.

②の成立：（Vx，Vai）∈E1，Y={A1，…，An}．FDHGの定義により，X→Ai∈F，Ai∈Y．故にArmstrongの合併律によりY∈X1.

nのときの成立を仮定する．

（n+1での①の成立）（Vx，Vy）∈E1(n)，（Vy，V_A）∈E1，帰納法の仮定によりY∈Xn．故にY→Aと合わせて，Armstrongの推移律により　A∈Xn+1.

（n+1での②の成立）（Vx，Vai）∈E1(n)，帰納法の仮定

よりAi∈Xn. Y= {A1, …, An} と合わせArmstrongの合併律により, Y∈Xn-1. □

次にFDHG, Hに対し同値な成分を定義し, それを用いてH上の擬閉路を定義する.

[同値な成分]

以下の(*)を満たすV∪HE1の部分集合Cを同値な成分とよぶ. 「Vx, Vy ∈C∩ (V∪HE1) ⇔ ^{T}H上で (Vx, Vy), (Vy, Vx) ∈^{T}E1」…(*). 上のCは, 一点v (∈V∪HE1) に対してCv= {v"∈V∪HE1 | (v, v"), (v", v) ∈^{T}E1} と一意に決り, v1∈CvならばCv1=Cvである.

[FDHG上の擬閉路]

同値な成分, C= {Vx1,…,Vxn} 上のハミルトン閉路 [17] をなす有向枝の集合 (Vx1, Vx2), (Vx2, Vx3),… (Vxn-1, Vxn), (Vxn, Vx1) ∈E1 (Vxi∈V∪HE1) をFDHG, H上の擬閉路cとよぶ. 以降, これをVx1⇒Vx2⇒…⇒Vxn⇒Vx1 とかく. Vx1, …, VxnはCに対して順序を除いて一意に決る. このとき, 命題6により以下が成立する.

[命題7] 次の(1)および(2)は同値である.

(1) FDHG, H上でVx, Vyは同一の擬閉路上に存在する.

(2) X←→Y∈F$^+$, (X, Y∈lhs (F))

3-3各Ri上の第三正規化に必要なFDHGi上の変形規則

次に, 章2および節4-1で定義した概念を用いて. 第三正規化の各過程と対応するFDHG上の変形を定義する. 但し, 複数の規則を組み合わせて正規化の一つの過程を構成する場合もある.

[変形規則]

規則1:擬閉路縮退

擬閉路cVa1⇒Va2⇒…⇒Van⇒Va1を一頂点V(a1)…(an)に縮退する. ここに, Va1は頂点または超枝. 更に, 縮退の際に擬閉路cに含まれる属性の1つを代表元として選択suる. 詳細には以下の操作を行う. c1, …, cnをFDHG, H上の擬閉路とする.

(1)各Ci:Vai⇒…⇒Van⇒Va1に対して, V(a1)…(an)を加える.

各V(a1)…(an)に対してその代表元Vai (∈V) を選ぶ.

(2)各Ci内のVaiに対して以下を行う.

(i) (he1, Vai) ∈E1" (he1∈HE1), (v, Vai) ∈E1' (v∈HE1∪V) を排除する,

(ii)he∩ (V−Ci) ≠ ϕ なるheに対してのみ (he, V(a1)…(an)) (ラベル1') をE1"に, v∈V−Ciなるvに対してのみ (v, V(a1)…(an)) (ラベル1") をE1'に加える. +-

(iii) (Vai, v) ∈E1'を削除し, v∈V−Ciなるvに対してのみ, (V(a1)…(an), v) (ラベル1) をE1'を加える.

(3) (出次数)^2+ (入次数)^2 = 0のV∪HEの元を削除.

(4)各Ci内のhe∈HE1に対して以下を行う.

(i) (he, v) ∈E1' (he∈HE1, v∈V) を削除し, (V(a1)…(an), v) (ラベル1) をE1'に加える.

(ii) (he, V(a1)…(an)) (ラベル1") (he∈HE1∩Ci) をE1'に加える.

上記の操作は, 関数従属性集合Fから候補キーをマージしたF'を導く部分に対応する.

規則2:閉包を求める

FDHG, H上に対し, 規則IおよびIIを適用することにより推移的閉包^{T}Hを求める. 命題6により, 規則2は {X→Y | X→Y∈F', X,Y ∈F_L∪F_R } を求める操作に対応する.

規則3:超枝内の冗長な頂点の排除

超枝he1(A1,A2,…, An) (he1(A1,A2,…, An),Van) ∈E1'に置いて, e∈^{T}(H−e) (eの始点はhe1, 終点はVai}) の場合にはVaiをhe1から排除し. eをE1から削除する. 命題6により, 規則3は, 関数従属性fの左辺lhs (f) が複数の属性から成る場合に, lhs (f) から導出型の冗長な属性を排除する操作に対応する.

規則4:推移的な有向枝の排除

規則2の閉包をとる操作によって他の有向枝から生成される有向枝の中で, 冗長な関数従属性に対応するものを排除する. 操作しては, ^{T}Hのラベル1+1"を持つ有向枝に対し, 手続き3 (付録) を適用し, 該当する有向枝を排除する.

規則5:関係スキームを示す超枝HE2の作成

各Va (∈V∪HE1) に対し, H上で以下を満たすVの部分集合をHE2に加える.

he2= {Var∈V | Var (Var∈V) を終点とし, Vaを始点とするラベル1の有向枝を持つ.} (但し, Va1が規則1により擬閉路が縮退されたものである場合には, その代表元をVarとして選ぶ)

規則6 孤立した超枝の排除

ラベル1の有向枝の始点にも終点にもなっていない超枝を削除する. 規則2, 規則4, 規則6の順に行うことで冗長な関数従属性を排除する部分に対応する.

3-4 INDとFDの相互作用によるFD導出に関する規則

節3. 2では一つの関係スキーム上のFDの導出および, その第三正規形データベーススキームへの変形をFD-ING-HG, H(F,I)の部分超グラフ上で行う手法を述べた. 本節では, 関係スキーム間制約であるINDと関係スキーム内での制約であるFDとの相互作用により導出されるFDを求めるためのFD-IND-HG, H(F,I)上の変形規則について述べる. 最初にI'に対応する, ^{T}E2について述べる.

[^{T}E2:E2の閉包]

以下の規則(1)を有限回適用して得られる, {(hei, hej:Ri∩Rj) | hei, hej∈HE2} 以下である最大の部分集合をE2の閉包と呼び, ^{T}E2とかく. 但し, ここで以下あるいは最大とは次の(*)による順序に基づくものとする.

(*) E2, E2'を共に, あるFD-IND-HGの, HE2の元をその始点および終点とする有向枝集合とする. ここでE2<E2'を「E2<E2'⇔hei, hej∈E2に対して, (hei,hej:X) ∈E2⇒ (hei,hej:Y) ∈E2' (Y⊇X) 」で定義する.

規則(1): (hei, hej:X), (hej, hek:Y) ∈E2, X⊆Y⇒ (hei, hek:X) をHE2に加える.

FDHGの場合の命題6と同様にして以下が言える.

[命題8]

Ri, Rj∈d (<R1, F1>, …, <Rn,Fn>, I) とするとき,

Ri[X]⊆Rj[X]∈I' ⇔ (he2(Rj), he2(Ri) : X) ∈^{T}E2. □

['E2:E2の簡約]

以下満たすE2の部分集合をE2の簡約とよび'E2とかく.

(i) ^{T} ('E2) =^{T}E2

(ii) E2' (⊆'E2), ^{T} ('E2) −^{T}E2'≠ ϕ なるE2'は存在しない.

FD-IND-HG, H(F,I)上でその簡約に関連する以下の操作を定義する.

規則(i):E2に対して簡約'E2を求める.

規則(ii):'H(F,I)= (V,HE1,HE2,E1,'E2) においてHE2の各元に, 'H(F,I)から導かれるG= (HE2, 'E2) の有向グラフにおける位相的順序 [21] を付与する. 以下これを単にHE2に位相的順序を付与すると呼ぶ.

更に，FDとINDの相互作用によるFDの導出律であるPull back 規則に対応する変形規則を設ける．
規則(iii)：（he2(Ri), he2(Rj)：X）∈E2かつ
(1) he1∪ {Va} ⊆∪$_{b\in x}$ {Vb} かつ，（he1, Va）∈E1|he2(Ri]（E1のhe2(Ri)への制限）のとき，he2(Rj)上で（he1, Va） をE1'へ，（he1, Vai）をE1"（Ai∈he1）へ加える．
(2) Va, Vc ∈∪$_{b\in x}$ {Vb} かつ（Va, Vc）∈E1|he2(Ri) のとき，he2(Rj)上で（Va, Vc）をE1'へ加える．
また，H(F,I)上の'E2に対応するIの簡約を定義する．
［Iの簡約I］
データベーススキームd=（<R1, F1>, ….<Rn, Fn>, I）に対応するFD-IND-HGをH=（V,HE1,HE2,E1,E2）とする．このときH'=（V,HE1,HE2,E1,'E2）に対応するデータベーススキームをd=（<R1, F1>, ….<Rn, Fn>, I'）とすると，I'をIの簡約とよび，I とかく．

4．IND-第三正規形関係スキーム作成手法

本章では，章3で定義した，FD-IND-HG，H(F,I)上の変形規則を組み合わせることにより，データベーススキームd=（＜R1, F1＞, …，＜Rn, Fn＞，I）を，(i)および(ii)を満たす第三正規形データベーススキームd'=（＜R'11, F'11＞, …，＜R'1j1, F'1j1＞, …，＜R'n1, F'n1＞, …＜R'njn, F'njn＞, I ）に変形するための手法を構成する．
(i) Ri＝R'i1∪…∪R'iji
(ii) $\overline{Fi}$=（F'i1∪…∪F'iji）$^+$
最初に，節4.1で各関係スキーム上Riに，関数従属性集合Fiが与えられた場合に（（<U,F>, ϕ）が与えられた場合を含む），それから第三正規形データベーススキーム作成を行う手法（手法2）を構成する．次に節4.2で，手法2をデータベーススキームd=（＜R1, F1＞, …，＜Rn, Fn＞，I）をIND第三正規形データベーススキームに変形する手法（手法3）に拡張する．

4-1関係スキーム上のFDHGを用いた正規化手法

Ri（U）上にFi（F）が与えられた場合に，冗長な関数従属性の排除を繰り返さない第三正規形データベーススキーム作成手法を構成する．
この場合に，後述する関数従属性に関する性質により，第三正規化で必要な各操作を一度限り行うためにはその適用順序が問題となる．本節では，Ri上のFD集合を単にFと記述する．
［関数従属性集合に関する性質］
① 左辺に導出型の冗長な属性を持つ関数従属性を含む関数従属性集合Fに対し，冗長な関数従属性の排除を行った後に冗長な属性の排除を行った場合には，冗長な関数従属性が排除されない場合がある[5, 18]
［例］F= {ABC→D, A→B, A→C, A→E, D→B} は導出型の冗長な属性BおよびCを左辺に持つ関数従属性ABC→Dを持つ．Fに対し，冗長な関数従属性の排除を行うと {ABC→D, A→B, A→C, A→E, D→B} を得るが，これに対し導出型の冗長な属性の排除を行うと {A→D, A→B, A→C, A→E, D→B} を得，冗長な関数従属性A→Bを排除することが出来ない．一方，先に冗長な属性の排除を行った場合には， {A→D, A→B, A→C, A→E, E→D} が得られ，これから冗長な関数従属性の排除を行うことで， {A→B, A→C, A→E, E→D} が得られ，冗長な属性および関数従属性は全て排除可能である．
② 導出型の冗長な属性の排除を行う以前に同値な候補キーのマージを行うと，導出型の冗長な属性は冗長な関数従属性を排除する段階では排除されない．従って，主キーに部分従属する非キー属性を含む関係スキーム作成を行う場合がある．
［例］関数従属性集合F＝{ABC→D, A→B, D→E, E→A, E→B, E→C, AC→F} では，Bは関数従属性ABC→Dの左辺において導出型の冗長な属性である．これを排除せずに，ABC←→Eに基づき両者をマージすると， {(ABC)(E)→D, B, C, A→B, D→E, AC→F} を得るが，これから冗長な関数従属性を排除しても導出型の冗長な属性Bは排除されず．キーABCに部分従属するD（F⊨{AC→D}）を持つ関係スキーム（(<u>(ABC)</u>(E),D）が作成される．
従って，導出型の冗長な属性の排除は，①により冗長な関数従属性の排除に先だって，また②により同値な候補キーのマージに先だって行う必要がある．また，冗長な関数従属性の排除を繰り返さないためには，冗長な関数従属性の排除に先立ち同値な候補キーのマージを行なわなければならない．従って，各過程は以下の順序で組み合わされなければならない．
［手法2：第三正規形関係スキーム作成手法］
step0 関数従属性の集合からFDHG,Hを作成
（この状態で，HE2＝E2＝ϕ）
step1 左辺から導出型の冗長な属性を排除
Hに規則3を適用し，H3を得る
step2 同値な候補キーをマージする．
H3に規則1を適用し，H31を得る
step3 冗長な関数従属性を排除する
1. 閉包を求める
H31に規則2を適用し，H312を得る．
2. 閉包を用いて，冗長なFDを排除する．
H312に規則4を適用し，H3124を得る．
H3124に規則6を適用し，H31246を得る．
step4 第三正規形関係スキームを作成
H31246に規則5を適用し，H312465を得る．
逆に，下記の定理1が示す様に，上記の手法を用いて構成された関係スキームは第三正規形である．
［定理1］上記の手法の手法によって構成された関係スキームは全て，第三正規形である．
（証明）step1はFの各元の左辺から導出型の冗長な属性を排除する操作に対応する．step2はFで同値なキーをマージしてF'を作る操作に対応するが命題4により，これにより新たに導出型の冗長な属性が生じることはない．step3により冗長な関数従属性を排除される（章2で述べた様に，step3終了時には導出型の冗長な属性も存在し得ない）．従って，step4で同一のキーを持つ属性をまとめて作成された関係スキームはキーに部分従属する非キー属性を持たず，また文献(2)の定理1．により，キーに推移従属する非キー属性も持たない．□
本手法は冗長な関数従属性の排除を同値な属性のマージの前後で繰り返さない．章2で示した，Berri and Bernsteinの手法では，冗長な関数従属性の排除を繰り返す例 {AB→C, AB→E,DE→A, DE→B, ACG→D,DE→F, F→C, F→H, C→G, H→C} に対しても，step1で導出型の冗長な属性を排除した後，step2で同値なキーAB，EFをまとめた後，step3で冗長な関数従属性の排除を一度限り行って，step 4 で第三正規形関係スキームが構成される．

4-2 FD－IND－HG上のIND第三正規形データベーススキーム作成法

最初に，Fi"＝F1⊔…⊔Fi-1⊔Fi+1⊔…Fnと I'との相互作用によって導出されるFDの集合 *Fi* をFiに加えることによ

り，各Ri上で，Fi∪*Fi*が与えられた条件のもとで第三正規形データベーススキームへの変換を行うことに帰着させる手法を以下に示す．

［手法３：IND第三正規化手法］

step0　データベーススキーム（＜R1,F1＞，…，＜Rn,Fn＞，　I）からFD-IND-HG，H(F,I)＝（V,HE1,HE2,E1,E2）を作成．

step1：Iのカバーリング[22]を求める．
　H(F,I)に規則(i)を適用し，E2に関する簡約'H(F,I)＝（V,HE1,HE2,'E1,E2）に変形する．

step2：Fi"とI'との相互作用によって導出されるFDの集合*Fi*をFiに加える．

step2-1　'H(F,I)に規則(ii)を適用して，HE2の各元に位相的順序を付与する．

step2-2　位相的順序の逆順に，各he2(Rp)上で次を行う．

step2-2-1　Hp='H(F,I)|he2(Rp)に規則３を適用して，he1∈Hpから冗長な頂点を削除する．

step2-2-2　FDHGpに規則２を適用して'Hpを求める．

step2-2-3　H(F,I)|(he2)（∃Xに対し（he2(Rp)，he2:X）∈'E2）と（he2，he2(Rp):Xi）に対して規則(iii)を適用する．

step3　各Hpに対して，手法２を適用して(Hp)312465を得る

［定理２］手法３は，DBS d＝（＜R1,F1＞，…＜Rn,Fn＞，I）が与えられたとき，IND第三正規形DBSd'＝（＜R'11,F'11＞，…，＜R'1i1, F'1i1＞，…＜R'nin, F'nin＞，I）で以下を満たすものを作成する．

(i)　Rq＝Rq1∪…∪Rqiq　（q＝1，…，n）

(ii)　$\overline{Fq}$＝（Fq1∪…∪Fqiq）$^{+}$　（q＝1，…，n）

（証明）命題２および定理１により，次の(1)，(2)を示せば十分である．

(1)　手法３により，Rq上に新たに課されるFD集合F(q)に対して，(*)……F(q)⊆*Fq*，F(q)$^{+}$＝*Fq*$^{+}$が成り立つ

(2)（＜Rq, Fq∪F(q)＞，φ）にBerri and Bernsteinの手法を適用すると，第三正規形DBS（＜Rq1，Fq1＞，…，＜Rqiq，Fqiq＞，　φ）で①Rq＝Rq1∪…∪Rqiq，②$\overline{Fq}$＝（Fq1∪…∪Fqiq）$^{+}$を満たすものが作成される．

（(2)の証明）命題２により，（Fq∪F(q)）$^{+}$＝（Fq∪*Fq*）$^{+}$を示せば十分である．（Fq∪F(q)）⊆（Fq∪*Fq*）より（Fq∪F(q)）$^{+}$⊆（Fq∪*Fq*）$^{+}$は自明．故に，(*)(*)…（Fq∪F(q)）$^{+}$⊇（Fq∪*Fq*）$^{+}$を言えばよい．F(q)$^{+}$＝*Fq*$^{+}$により（Fq^{+}∪F(q)$^{+}$）$^{+}$＝（Fq^{+}∪*Fq*$^{+}$）$^{+}$⊇（Fq∪*Fq*）$^{+}$．更に，Fq^{+}∪F(q)$^{+}$⊆（F∪F(q)）$^{+}$であるので，（Fq^{+}∪F(q)$^{+}$）$^{+}$⊆（（Fq∪F(q)）$^{+}$）$^{+}$が成り立つが，関数従属性集合の閉包の定義により（（Fq∪F(q)）$^{+}$）$^{+}$＝（Fq∪F(q)）$^{+}$であるので（Fq^{+}∪F(q)$^{+}$）$^{+}$⊆Fq∪F(q)）$^{+}$である．以上により(*)(*)が成り立つ．

（(1)の証明）以下，he2（Rq）をhe2（q）と略記する．he2（q）で'E2の元の（入り次数）＊（出次数）≠０の元をhe2(1)，…，he2(N)と仮定してよい．そうでない場合にはRiの添字を付け代えればよい．今，he2(q)（q=1,…,N）は位相的順序の逆順に整列されていると仮定してよい．実際，もしそうでない場合には，各関係スキームRqの添字を付け代えて整列させることが出来るからである．以下HE2'＝{he2(1)，…，he2(N)}の各元に対応するRqに関して，手法３stepIIが終了した時点でRqに新たに加えられた関数従属性の集合F(q)（則ち，手法３stepIIが終了した時点でRq上に課されたている関数従属性集合をFq"としたとき，F(q)=Fq"－Fq）が(*)を満たすことをqに関する帰納法で示す．

初期段：he(1)を終点とするE2の元は存在しない．故に手法３の適用によりR1に新たに加えられる関数従属性は存在せずF(1)＝φである．また，Rj[X]⊇Rq[X]となるj，Xは存在しない．HE2'の元はG=（HE2'，'E2）における位相的順序の逆順に整列されているので，Rj[X]⊇R1[X]を満たすj，Xは存在しない．よってpull back規則を適用することによりR1上に導かれる関数従属性は存在しないため*F1*＝φであり，結局F(1)=*F1*＝φが成立し，q=1の場合には(*)が成り立つ．

帰納段：{he2(1)，…he2(N)}は位相的順序の逆順に整列されているので，<u>he2(m)</u>＝∪$_{X}$ {he2∈HE2 | X⊆Rm+1，（he2，he2(m+1):X）∈'E2}⊆{he2(1)，…he2(N)}．また「$^{\forall}$he2(r)（$^{\exists}$Xについて（he2(r)，he2(m+1):X）∈'E2）について$^{\exists}$Y（⊇X），1≦$^{\exists}$j≦m，$^{\exists}$he2(j)∈<u>he2(m+1)</u>に対して，（he2(r)，he2(j):X）∈'E2かつ，（he2(j)，he2(m+1):Y）∈'E2」…(*)(*)．これは，(*)(*)と同じY，jに対して，Rj[X]⊆Rq[X]∈I'，Rm+1[Y]⊆Rj[Y]∈*I*が成り立つことを意味する．故に，Fq^{+}の元と包含従属性Rm+1[X]⊆Rq[X]（∈I'）に対し，pull back規則を適用することによりRm+1上に加えられる関数従属性の集合F(q→m+1:X)は，（Fj∪*Fj*）$^{+}$の元および包含従属性Rm+1[Y]⊆Rj[Y]（∈*I*）に対しpull backを適用することによりRm+1上に加えれる関数従属性の集合に含まれる．ここで，j≦mより帰納法の仮定からF(j)$^{+}$＝*Fj*$^{+}$であるので，（Fj∪F(j)）$^{+}$＝（Fj∪*Fj*）$^{+}$．故に，F(q→m+1:X)は（Fj∪F(j)）$^{+}$の元および包含従属性Rm+1[Y]⊆Rj[Y]に対しpull back規則を適用することによりRm+1上に加えられる関数従属性の集合に含まれる．故に手法３のstep2-2およびF(m+1)の定義により，F(q→m+1:X)⊆F(m+1)$^{+}$．上記がRj[X]⊇Rm+1[X]を満たす全てのRjおよびXについて成り立つのでF(m+1)$^{+}$⊇*Fm+1*．故に，F(m+1)$^{+}$＝(F(m+1)$^{+}$)$^{+}$⊇*Fm+1*$^{+}$．またF(m+1)の作り方より明らかにF(m+1)⊆*Fm+1*．故にF(m+1)$^{+}$⊆*Fm+1*$^{+}$でもあり，F(m+1)$^{+}$＝*Fm+1*$^{+}$が成立する．以上でq=m+1の場合の(*)の成立が言える．□

4-3　本手法の時間的複雑さ

手法３を用いてd＝（＜R1，F1＞，…，＜Rn，Fn＞，I）からIND第三正規形DBSを作成する場合および文献(1)の（＜U,F＞,φ）から第三正規形DBSへの変換手法の各stepの時間的複雑さを表1に示す．特にd＝（<U,F>，I）の場合には，全体として従来手法と同等の時間量を要するが，非導出型の冗長な属性を排除する部分では従来手法に比して有効であると考えられる（表1，参照）

表１　本手法の時間的複雑さ

		Berri and Bernstein	本手法:括弧内はd=(<U,F>,φ)の場合
Iの簡約'Iを求める		考慮せず	O(i²)（無し）
各Riに'Iの元との相互作用により導かれるFDを加える			O(i·L²)（無し）
冗長な属性の排除	導出型の冗長な属性の排除	O(L²)	O(nL²)（O(L²)）
	非導出型の冗長な属性排除		O(nP·L)（O(p'·L')）
同値なキーのマージ		O(p·L)	O(nL²)（O(L²)）
冗長なFD排除		O(p·L)	O(nPL)（O(p'L')）
全体		O(L²)	O(L²·max(i,n))（O(L²)）

p=|F|，p'=|F'|，L＝len（F），L'＝len（F'），p'·L'≦L²，i＝|I|，P=|F|（F=F1⊔…⊔Fn），L＝len（F）

5．むすび

本稿では，属性の全体集合上の普遍関係を必ずしも仮定せず，各関係スキーム内の制約である関数従属性（FD）と関係スキーム間の制約である包含従属性（IND）を与えた場合に，FDおよびIND固有の導出律のみならず，これらの相互作用に対応する導出規則であるpull back規則をも対象とした場合の第三正規形データベーススキーム設計手法を提案した．本手法は普遍関係上の関数従属性集合に対しては，従来の第三正規化手法と同様にCoddのいう第三正規形データベーススキームを作成する．則ち，提案手法は従来の第三正規化手法の拡張形と考えられる．

また，本手法は各関係スキーム（普遍関係スキームUの場合を含む）とその上の関数従属性集合から第三正規形データベーススキーム作成を行う場合に，従来のBerri and Bernsteinの手法[1]等と異り，冗長な関数従属性の排除を同値なキーのマージの前後で繰り返さない．

謝辞本検討を進めるに当りご支援頂いた，当研究所の石垣データベース研究部長に感謝致します．

参考文献

(1) Berri, C. and Bernstein, P.A. : Computational Problem Related to the Design of Normal Form Relational Schemes, *ACM Trans. Database Syst.*, Vol. 4, No. 1, pp. 30-59 (1979).

(2) Bernstein, P.A. :Synthesizing third normal form relations from functional dependencies, *ACM Trans. Database Syst.*, Vol. 1, No4, pp. 272-298 (1976).

(3)Ausiello, G., Da'tri,A and Sacca, D. : Graph Algorithms for Functional Dependency Manipulations, *J.ACM*, Vol. 30, No. 4, (1983).

(4) Maier, D.: Minimum Covers in the Relational Database model, *J.ACM*, Vol. 27, No. 4, pp. 664-674 (1980).

(5) Diederich, J. and Milton, J.: New Methods and Fast Algorithms for Database Normalization, *ACM Trans. Database Syst.*, Vol. 13, No. 3, pp. 339-365 (1988).

(6)Teorey, J.T., Yang, D. and Fry, P.F. :A Logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model., *ACM Computing Surveys*, Vol. 18, pp. 197-228(1986).

(7)Hull, R. and King, R. :Semantic Database Modeling:Survey, Applications and Research Issues., *ACM Computing Surveys*, Vol. 19, No. 3, pp. 201-260(1987).

(8)Abiteboul, S. and Hull, R. :IFO: A Formal Semantic Model., *ACM Trans. Database Syst.*, Vol. 12, No. 4, pp. 525-565(1987).

(9)Jajodia, S., Ng, P.A. and Springsteel, F.N. :The Problem of Equivalence for Entity-Relationship Diagrams, *IEEE Trans. Softw. Eng.*, Vol. 9, No. 5, pp.617-630(1983).

(10)Markowitz, V.M., and Shoshani, A. :On the Correctness of Representing Extended Entity-Relationship Structure in The Relational Model., proceedings of the 1991 ACM SIGMOD International Conference on Management of Data, pp.430-439(1989).

(11)Makowsky, J.A., Markowitz, V.M. and Rotics, N. :Entity-Relationship Consistency for Relational Schemas, Proceedings of International Conference on Database Theory, pp.306-322(1986).

(12)Casanova, M.A., Fagin, R. and Papadimitriou, C.H. : Inclusion Dependencies and Their Interaction with Functional Dependencies, Proceedings of the 1st ACM SIGACT-SIGMOD Symposium on Principle of Database Systems, pp. 171-176(1982).

(13)Casanova, M.A., Fagin, R. and Papadimitriou, C.H. : Inclusion Dependencies and Their Interaction with Functional Dependencies, *J. Comput. Syst. Sci.* Vol. 28,pp.29-59(1984).

(14)Ling, T.W. and Goh, C.H.:Logical Database Design with Inclusion Dependencies, proceedings of the 8th International Conference on Data Engineering, pp. 642-649(1992).

(15)Cosmadakis, S.S. and Kanellakis, P.C. :Functional and Inclusion Dependencies A Graph Theoretic Approach, Proceedings of the ACM symposium on Theory of Computing, pp.29-37(1984).

(16)Cosmadakis, S.S., Kanellakis, P.C. and Vardi, M.S. :Polynomial-Time Implication Problems for Unary Inclusion Dependencies, *J. ACM*, Vol. 37, No.1, pp. 15-46(1990).

(17)Atzeni, P. and Chan E.P.F. : Independent Database Schemes under Functional and Inclusion Dependencies, Proceedings of the 13th International Conference on Very Large Data Bases, pp. 159-166(1987).

(18) Ullman, J.D. :Principles of Database and Knowledge-base Systems, Vol. I, Computer Science Press, Rockville, MD (1988).

(19) Berge, C. : Graphs and Hypergraphs, p. 389, North-Holland, Amsterdam, (1973).

(20) Aho, A. V., Hopcroft, J. E. and Ullman, J.D. : Data Structures and Algorithms, Addison-Wesley, (1983).
（邦訳，大野義男訳：データ構造とアルゴリズム，培風館，p. 186，東京(1987).）

(21)Tarjan, R.E. :Data Structures and Network Algorithms, the Society for Industrial and Applied Mathematics.(1983).（邦訳，岩野和男訳：データ構造とネットワークアルゴリズム，pp. 30～31，マグロウヒル，東京(1989).）

付録．冗長なFDに対応する有向枝を排除する手法

① ラベル1+1'の有向枝e∈E1を削除する有向枝の候補として選択する．

② ①で選ばれた有向枝eのうち，命題5の条件(1)を満たすものをE1から排除する．

③ ①で選ばれ②で選ばれなかった有向枝eのうち，付補題2の条件(4)を満たすものをE1から排除する．

［閉包を求めるための手法］

T_I，T_{II}の定義により，$T_{II}^2(E1)=T_{II}(E1)$，$T_IT_{II}(E1)\supseteq T_I(E1)$，$T(E1)\supseteq E1$．故に$^\forall\gamma\in\Gamma$に対して，$|\gamma|=n+m$（$\gamma(x,y)=x^n y^m$）とすると，$\gamma(T_I, T_{II})(E1)\subseteq(T_IT_{II})^{|\gamma|}(E1)$．故に，$^TE1\subseteq\cup_k(T_IT_{II})^{|k|}(E1)$．ところが$^\forall k\in N$に対して，$(T_IT_{II})^{|\gamma|}(E1)\subseteq E1^*$．但し$E1^*=\{(he,v) \mid he\in HE1,\ v\in V-he$，(he,v)のラベル1+1"$\}\cup\{(v,he) \mid v\in V-he$，(v,he)のラベル1"$\}\cup\{(he,w) \mid he\supseteq w\ (w\in HE1),\ he\ni w\ (w\in V)\}$

故に，$\cup_k(T_IT_{II})^{|k|}(E1)$は収束し，$|(T_IT_{II})^{|k|}(E1)|\leq|E1^*|<\infty$であるので，あるk'に対しk≧k'で$(T_IT_{II})^k(E1)=(T_IT_{II})^{k'}(E1)$．故に，$\cup_k(T_IT_{II})^{|k|}(E1)=(T_IT_{II})^{k'}(E1)={}^TE1$である．今，$w\in HE1\cup V$（wの出次数≧1）に対し，$W_0=\{w\}$，$W_{i+1}=\{(w,v) \mid (T_IT_{II})(E1\cup W_i)\ni(w,v)\}$で$\{W_i\}_{i=1,2,\ldots}$を定義する．このとき$W_0\subseteq W_1\subseteq\cdots$，かつ$W_i\subseteq{}^TE1$より，$\lim_{i\to\infty}W_i$が存在する．ここで$|W_i|\leq|{}^TE1|<\infty$なので，$^\exists i_w$，$^\forall i(\geq i_w)$に対して，$W_i=W_{i_w}$（$i\geq i_w$）．故に，各wについて$W_{i_w}\subseteq{}^TE1$．結局，$\cup_{w\text{の出次数}\neq0}W_{i_w}\subseteq{}^TE1$．$W_i$の定義から，$(T_IT_{II})^k(E1)\subseteq\cup_{w\text{の出次数}\neq0}W_{i_w}$．故に，${}^TE1=\cup_{w\text{の出次数}\neq0}W_{i_w}$．よって，以下の手続きにより$^TE1$が得られる．

step1 各w（∈V∪HE1，wの出次数≠0）に対してW_{i_w}を求める．