REIHE INFORMATIK

2/1990

Ensuring the Existence of a BCNF-Decomposition that preserves Functional Dependencies in O (N^2) Time

Mila E. Majster-Cederbaum

Fakultät für Mathematik und Informatik

Universität Mannheim

Seminargebäude A5

6800 Mannheim

1. INTRODUCTION

In the design theory for the relational data base model [5] the decomposition of a relation scheme according to some normal form has been advocated for as one means to avoid undesired redundancy and anomalies [6, 7, 15]. Desired characteristics of such a decomposition in the case of a relation scheme R with an associated set F of functional dependencies are the lossless-join property and the preservation of functional dependencies [12, 15]. Every relation scheme R with associated set F of functional dependencies has a decomposition into third normal form (3NF) that has the lossless-join property and preserves functional dependencies. However, for relation scheme R with associated set F of functional dependencies, a decomposition into Boyce-Codd normal form (BCNF) that has the lossless-join property and preserves functional dependencies does not always exist. Beeri and Bernstein [2] have shown, that, given a relation scheme R and a set F of functional dependencies, i) deciding if a subscheme S of R is in BCNF is NP-complete, ii) deciding if R has a BCNF-decomposition that preserves functional dependencies is NP-complete. Osborn [13] and Le Doux and Parker [11] give algorithms for determining, if, for given R and F, R has a BCNF decomposition preserving functional dependencies, that use exponential time in the worst case. Isou and Fischer [14] give a polynomial-time algorithm for finding a BCNF-decomposition that has the lossless-join property, but the decomposition may not preserve functional dependencies.

We are here presenting a simple condition (which can be tested in $O(N^2)$ time) that ensures that a relation scheme R with set F of functional has a BCNF- decomposition that has the lossless-join property and preserves functional dependencies .

2. DEFINITIONS AND ELEMENTARY PROPERTIES

A <u>relation scheme</u> R is a finite set of attribute names, $R = \{A_1, \ldots, A_n\}$. Associated with each attribute name A_i is a set D_i, $1 \le i \le n$, called the <u>domain</u> of A_i. Let $D = D_1 \cup D_2 \cup \ldots \cup D_n$. A <u>relation</u> r <u>on</u> the relation scheme R is a finite set of mappings, $r = \{t_1, \ldots, t_n\}$, from R to D with the restriction that for each $t \in r$, $t(A_i)$ must be in D_i, $1 \le i \le n$. The t_i are called <u>tuples</u>.

Let X, Y ⊂ R. A functional dependency (FD) for R is an expression f of the form X -> Y. X is called the left-hand side of f and Y is called the right-hand side of f.

A relation r on R satisfies the FD X -> Y, if for any two tuples t_1, t_2 ∈ r $t_1(X) = t_2(X)$ implies that $t_1(Y) = t_2(Y)$. Here, $t_1(X)$ denotes the image of X under t_1, i. e. $t_1(X) = \{t_1(A) : A \in X\}$.

Let F be a set of FDs for R. The pair (R,F) is called a relational description. A relation r on R is a relation for (R,F), if r satisfies every f ∈ F.

Let (R,F) be a relational description, f = X -> Y a FD for R. F logically implies f, written F ⊨ f, if every relation r for (R,F) also satisfies f. The set of all FDs that are logically implied by F is denoted by

$$F_R^+ = \{X \to Y \mid F \models X \to Y\} .$$

Given (R,F) we define the closure of F with respect to R as the smallest set CL (R,F) of functional dependencies for R that satisfies the following conditions

1) F ⊆ CL(R,F)
2) if Y ⊆ X ⊆ R then X -> Y ∈ CL(R,F) (reflexivity)
3) if X -> Y ∈ CL(R,F) and Z ⊆ R then
XZ -> YZ ∈ CL(R,F), where XZ denotes X ∪ Z (augmentation)
4) if X -> Y ∈ CL(R,F) and Y -> Z ∈ CL(R,F) then
X -> Z ∈ CL(R,F) (transitivity)

Conditions 2) to 4) are often referred to as Armstrong's rules [15].

It has been shown [1, 9, 12, 15] that

$$CL(R,F) = F_R^+ .$$

This means in particular that every FD f ∈ F_R^+ can be obtained from F by repeatedly applying rules 2) to 4). (*)

(*) We omit the subscript R, if no ambiguity arises

Given (R,F) and $X \leq R$ one defines

$$X^+_{(R,F)} = \{A \in R : X \to A \in F^+_R\}.$$

$X^+_{(R,F)}$ can be easily calculated as follows, see [3, 12, 15]

$X^{(0)} = X$

$X^{(i+1)} = X^{(i)} \cup \{A \in R : \exists f = Y \to Z \in F$ such that $Y \leq X^{(i)}$ and $A \in Z\}$

$X^+_{(R,F)} = \bigcup_{i \geq 0} X^{(i)}$

It is easily seen, e.g. [15], that

$$X \to Y \in F^+_R \text{ iff } Y \leq X^+_{(R,F)} \;.^{(*)}$$

This observation yields an algorithm, that, given a relational description (R,F) with $|R| = n$ and $|F| = m$ and a FD f for R tests in $O(nm^2)$, if $f \in F^+_R$, see e.g. [12].

Given (R,F) and (R,G), F and G are called equivalent, written $F \equiv G$, if $CL(R,F) = CL(R,G)$. If $F \equiv G$, we say F is a cover for G (and vice versa).

A set F of FDs for R is nonredundant if there is no proper subset F' of F with $F' \equiv F$. F is a nonredundant cover for G, if it is a cover and nonredundant. A FD $X \to Y$ in F is called redundant if $(F \setminus \{X \to Y\}) \models X \to Y$. Clearly, F is nonredundant iff no $f \in F$ is redundant.

Given (R,F) and $X \to Y \in F$ and $A \in R$, we say that A is extraneous in $X \to Y$ if

1) $X = AZ$, $X \neq Z$, and $(F \setminus \{X \to Y\} \cup \{Z \to Y\}) \equiv F$

or

2) $Y = AW$, $W \neq Y$, and $(F \setminus \{X \to Y\} \cup \{X \to W\}) \equiv F$

$X \to Y$ is called leftreduced if X contains no attribute extraneous in $X \to Y$.

$X \to Y$ is called rightreduced if Y contains no attribute extraneous in $X \to Y$.

$X \to Y$ is called reduced, if it is leftreduced and rightreduced and $Y \neq \emptyset$.

(*) We omit the subscript (R,F) if no ambiguity arises

A set F of FD is called left-reduced (right-reduced, reduced), if every $f \in F$ is left-reduced (right-reduced, reduced).

F is called canonical, if every $f \in F$ is of the form $X \rightarrow A$, $A \in R$, and F is leftreduced and nonredundant. F is a canonical cover for G, if F is canonical and a cover. Clearly, if F is canonical then F is reduced. Also, if F is reduced then taking each FD $X \rightarrow A_1 \dots A_m$ in F and splitting it into $X \rightarrow A_1, \dots , X \rightarrow A_m$ yields a set G of FD that is a canonical cover for F, see e.g. [12]. For a given relational description (R,F) a reduced cover, and hence a canonical cover, can be determined in $O(n^2\ m^2)$ time, where $|R| = n$ and $|F| = m$.

A relational description (R,F) is in Boyce - Codd normal form (BCNF), if for every $X \subseteq R$

$$(X \rightarrow Y \in F^+ \text{ with } Y \not\subseteq X \text{ implies } X \rightarrow R \in F^+)$$

[6, 7]. There is a polynomial-time algorithm that decides, if (R,F) is in BCNF [13], but it is NP-complete [10] to decide for (R,F) and $S \subseteq R$ if S is in BCNF with respect to F [2].

A decomposition for (R,F) is a set $\rho = \{R_1, \dots R_k\}$ of relation schemes R_i such that

$$\bigcup_{i=1}^{k} R_i = R.$$

A decomposition $\rho = \{R_1, \dots , R_k\}$ for (R,F) has the lossless-join property, if for every relation r for (R,F)

$$\Pi_{R_1}(r)|x|\Pi_{R_2}(r)|x|\dots|x|\Pi_{R_k}(r) = r$$

Given a decomposition $\rho = \{R_1, \dots , R_k\}$ for (R,F) the projected dependencies are defined as

$$F_i = \{f \in F^+ : f = X \rightarrow Y \text{ and } X\,Y \subseteq R_i\}$$

$1 \leq i \leq k$.

A decomposition $\rho = \{R_1, \dots , R_k\}$ for (R,F) is a BCNF-decomposition, if every (R_i, F_i) in BCNF.

A decomposition $\rho = \{R_1, \dots, R_k\}$ for (R,F) is said to <u>preserve functional dependencies</u> (or to be a covering decomposition) if

$(\bigcup_{i=1}^{k} F_i) \equiv F$, i.e. if $\bigcup_{i=1}^{k} F_i$ covers F.

It is well-known that for every (R,F) there exists a decomposition $\rho = \{R_1, \dots, R_k\}$ such that

- ρ has the loss-less join property
- ρ preserves functional dependencies
- (R_i, F_i) is in third normal form, $1 \le i \le k$, see [2, 3, 4, 8, 12, 15].

In addition, it is known, that for every (R,F) there exists a decomposition $\rho = \{R_1, \dots R_k\}$ such that

- ρ has the loss-less join property
- ρ is a BCNF-decomposition

see [12, 14, 15].

On the other hand, it is known that there are relational descriptions for which no decomposition exists that preserves functional dependencies and is a BCNF decomposition. A "minimal" standard example for such a relational description is

$$R = ABC$$
$$F = \{AB \to C,\ C \to A\}.$$

It is NP-complete to decide if a given (R,F) has a functional dependency preserving BCNF decomposition [2].

In [13] Osborn introduces the concept of a covering BCNF for (R,F). A covering BCNF for (R,F) is a set $\{(R_1, G_1), \dots (R_\ell, G_\ell)\}$ such that

- each G_j contains all functional dependencies in $\bigcup_{i=1}^{\ell} G_i$ over R_i

- each R_j is in BCNF according to $\bigcup G_i$

- $\bigcup G_i$ is a cover for F.

Osborn presents an algorithm that tests for a given (R,F), if there exists a covering BCNF for (R,F) in $O(n^3 m\ 2^{2n})$, where $|R| = n$ and $|F| = m$. It is easy to see that a BCNF-decomposition that preserves functional dependencies is a covering BCNF. On the other hand every covering BCNF can be used to construct a BCNF decomposition that preserves functional dependencies. Hence, the question, if a relational description (R,F) has a functional dependencies preserving BCNF-decomposition can be answered in $O(n^3 m\ 2^{2n})$.

Example 1

Let R = ABCD F = {B -> C}. Let R_1 = BC and G_1 = {B -> C} then $\tau = \{(R_1, G_1)\}$ is a covering BCNF.
Let $\rho = \{R_1, R_2\}$, where R_2 = AD and F_j, j = 1, 2, as defined before, then ρ is a BCNF-decomposition that preserves functional dependencies.

A BCNF-decomposition for (R,F) that preserves functional dependencies may fail to possess the loss-less join property, as is easily seen from examples 1 and 2.

Example 2

R = ABCD
F = {AB -> C, CD -> A}
ρ = {ABC, ACD} is a BCNF-decomposition that preserves functional dependencies but does not possess the loss-less join property.

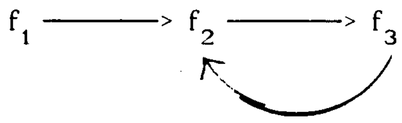
Let (R,F) be a relational description, such that every $f \in F$ is of the form $X \rightarrow A$, $A \in R$.
We associate with (R,F) a directed graph G(R,F) : the nodes of G(R,F) are the functional dependencies in F. Whenever $f_1 = X_1 \rightarrow A_1$ and $f_2 = X_2 \rightarrow A_2 \in F$ and $A_1 \in X_2$ then we draw an edge from f_1 to f_2.

Example 3

R = ABCDEGH

$F = \{f_1 = AB \text{ -> } C,\ f_2 = CD \text{ -> } E,\ f_3 = EHG \text{ -> } D\}$

has the graph



2. A CONDITION FOR THE EXISTENCE OF A BCNF-DECOMPOSITION THAT HAS THE LOSSLESS-JOIN PROPERTY AND PRESERVES FUNCTIONAL DEPENDENCIES

In this section we want to establish a simple condition that ensures for (R,F) the existence of a BCNF-decomposition that preserves functional dependencies and has the lossless-join property.

For this we need the following auxiliary lemmata.

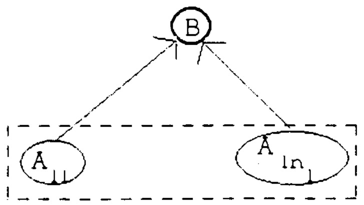
Lemma 1
Let (R,F) be a relational description such that every $f \in F$ is of the form $X \text{ -> } A$, $A \in R$. Let $g = B_1 \dots B_m \text{ -> } B \in F^+_R$, $B \neq B_i$, $i = 1 \dots m$. If, for some j, B_j is not extraneous in g, then there exist FDs f_1, f_2 in F such that

i) B_j is contained in the lefthandside of f_1
ii) B is the righthandside of f_2
iii) there is a directed path from f_1 to f_2 in G(R,F)

Proof:
We observe that, under the conditions of the lemma, for any $X \leq R$

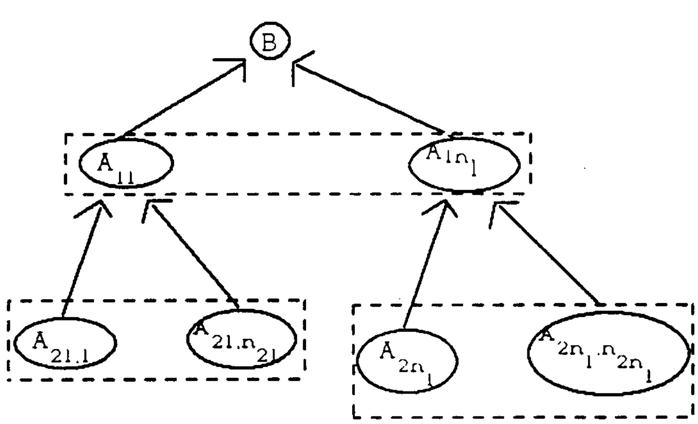
$$X^+ = \bigcup X^{(i)}, \text{ where}$$
$$X^{(o)} = X$$
$$X^{(i+1)} = X^{(i)} \cup \{A \in R : \exists\, \alpha \leq X^{(i)} \text{ such that } \alpha \text{ -> } A \in F\}$$

As $B_1 \dots B_m \text{->} B \in F^+$ iff $B \in (B_1 \dots B_m)^+$, we immediately obtain that there must be some FD f that has B as its right-hand side. Let now B_j be not extraneous in g. As $B \in (B_1 \dots B_m)^+$ there must be some smallest integer k such that $B \in (B_1 \dots B_m)^{(k)}$. Let $f_2 = \alpha_1 \rightarrow B$ be the FD(*) that was applied to include B into $(B_1 \dots B_m)^{(k)}$, then $\alpha_1 \subseteq (B_1 \dots B_m)^{(k-1)}$. We depict this situation graphically by creating a node with label B and a node with label A for every attribute A of α_1 and connect these nodes as follows



$$f_2 = \alpha_1 \longrightarrow B$$

where $\alpha_1 = A_{11} \dots A_{1n_1}$. Each A_{1j} is in $(B_1 \dots B_m)^{(k-1)}$ and hence A_{1j} was obtained by some FD $\alpha_{2j} \rightarrow A_{1j}$, i.e.



$\alpha_1 = A_{11} \dots A_{1n_1}$

$\alpha_{21} = A_{21,1} \dots A_{21,n_{21}}$ etc.

(*) There may be more than one such FD, we choose one.

Please note that different nodes may carry the same label. We continue this process and finally after maximal k steps we end up with a situation where the leaves of the tree such constructed are labelled with elements of $B_1 \ldots B_m$ belonging to some left-hand-side of some functional dependencies in F. Let $b_1 \ldots b_r$ be the labels of the leaves. Let us assume that none of the $b_1 \ldots b_r$ is B_j. We then consider the set

$$b_1 \ldots b_r \subseteq B_1 \ldots B_m .$$

Clearly $b_1 \ldots b_r \rightarrow B \in F^+$ and $B_j \not\subseteq b_1 \ldots b_r$ hence B_j is extraneous in g yielding a contradiction. Hence we conclude that there is some b_{j_o} with $B_j = b_{j_o}$. Let f_1 be the FD that contains $B = b_{j_o}$ in its left-hand-side (choose one, if there is more than one). Then by the above construction and by the definition of G(R,F) there is a path from f_1 to f_2 in G(R,F).

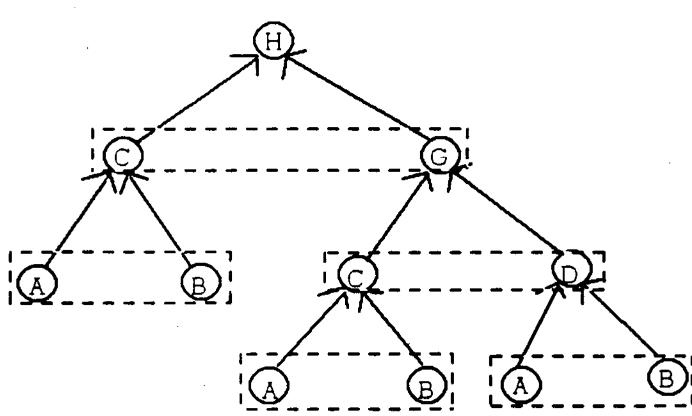
Example 4

R = ABCDGHLM

$F = \{f_1 = AB \rightarrow D,\ f_2 = CG \rightarrow H,\ f_3 = CD \rightarrow G,\ f_4 = AB \rightarrow C\}$

$(AB)^{(0)} = AB$, $(AB)^{(1)} = ABCD$, $(AB)^{(2)} = ABCDG$

$(AB)^{(3)} = ABCDGH$, hence $AB \rightarrow H \in F^+$, where B is not extraneous and a tree constructed according to Lemma 1 is



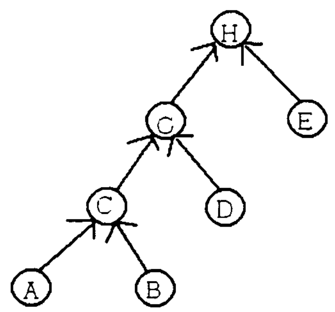
Choosing the rightmost occurence of B we obtain the path $f_1 \rightarrow f_3 \rightarrow f_2$.

Example 5

R = ABCDGH

F = {f_1 = CD -> G, f_2 = GE -> H, f_3 = AB -> C}

then ABDE -> H $\in$ F^+ and D is not extraneous. A tree constructed according to the theorem is



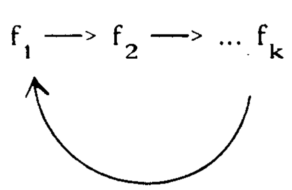
from where we get the path consisting of one edge f_1 -> f_2.

Lemma 2

Let (R,F_1) and (R,F_2) be canonical and $F_1 \equiv F_2$. $G(R,F_1)$ has a directed cycle iff $G(R,F_2)$ has a directed cycle.

Proof:

Let $G(R,F_1)$ have a directed cycle. Let w.l.o.g. the cycle be



with $f_i = X_i \rightarrow A_i$, $X_i \subseteq R$, $A_i \in R$, $1 \leq i \leq k$.

Hence we conclude that

$$A_i \in X_{i+1} \qquad\qquad i = 1, \dots k-1$$

and

$$A_k \in X_1.$$

As $f_1 \in F_2^+$ and f_1 is leftreduced we know by Lemma 1 that there are FD g_{11}, g_{12} in F_2 such that

- A_k is contained in the left-hand-side of g_{11}

- A_1 is the right-hand-side of g_{12}

and there is a path from g_{11} to g_{12} in $G(R,F_2)$, i.e.

$$g_{11} \rightsquigarrow g_{12}.$$

As $f_2 \in F_2^+$ and f_2 is leftreduced and $A_1 \in X_2$ we conclude that there are FDs g_{21}, g_{22}, $\in F_2$ such that

- A_1 is contained in the left-hand-side of g_{21}

- A_2 is the right-hand-side of g_{22}

and there is a path from g_{21} to g_{22} in $G(R,F\ )$. Now, clearly there is an edge from g_{12} to g_{21} in $G(R,F_2)$, so we get

$$g_{11} \rightsquigarrow g_{12} \longrightarrow g_{21}, \rightsquigarrow g_{22}.$$

Continuing like this we get a path

$$g_{11} \rightsquigarrow g_{12} \longrightarrow g_{21} \rightsquigarrow g_{22} \longrightarrow g_{31} \cdots g_{k-1,2} \longrightarrow g_{k,1} \rightsquigarrow g_{k,2}$$

where the path between $g_{\ell,1}$ and $g_{\ell,2}$ is obtained from the fact that $f_\ell \in F_2^+$, $\ell = 1 \dots k$.

As there is an edge from $g_{k,2}$ to g_{11} in $G(R,F_2)$, we found a directed cycle in $G(R,F_2)$.

Lemma 3

Let (R,F) be a relational description.

i) Let $\rho = \{R_1, R_2\}$ be a decomposition for (R,F).
ρ has the lossless-join property iff $(R_1 \cap R_2) \longrightarrow (R_1 \setminus R_2) \in F^+$ or $(R_1 \cap R_2) \longrightarrow (R_2 \setminus R_1) \in F^+$

ii) if (R,F) is not in BCNF, then there are attributes A, B, $A \neq B$, $\in R$ such that $(R \setminus \{A,B\}) \longrightarrow A \in F^+$.

Proof:

i) this proof can be found e.g. in [12, 15]

ii) this proof can be found e.g. in [15]

Theorem

Let (R,F) be a relational description, F canonical. If G(R,F) does not contain a directed cycle, then there exists a BCNF-decomposition of (R,F) that has the lossless-join property and preserves functional dependencies.

Proof:

Let $F = \{f_1 = X_1 \longrightarrow A_1, \dots , f_n = X_n \longrightarrow A_n\}$.

Consider the decomposition

$$\rho = \{R_1, \dots, R_n, R_{n+1}\}$$

where $R_i = X_i A_i$ $\qquad i = 1 \dots n$

$$R_{n+1} = R \setminus \{A_1, \dots , A_n\}.$$

R_{n+1} may be empty. We observe that $R_i \neq R_j$ for $i \neq j$ under the condition of the theorem. Clearly, ρ is a decomposition of (R,F). Moreover, it is obvious that ρ preserves functional dependencies. We have to show that each subscheme R_i is in BCNF with respect to the respective projection F_i of F^+. It is clear that R_{n+1} is in BCNF, because any nontrivial FD in F^+ must involve some A_j.

Let us now consider $R_1 = X_1A_1$. We assume that (R_1,F_1) is not in BCNF. Then by Lemma 3 ii) there are A, B, $A \neq B, \in R_1$ such that

$$(R_1 \setminus \{A,B\}) \longrightarrow A \in F_1^+ \subseteq F^+ .$$

Let $X_1 = B_1 \ldots B_m$. $B_i \in R$, i.e. we get

$$(B_1 \ldots B_m A_1 \setminus \{A,B\}) \longrightarrow A \in F^+ .$$

We can distinguish the following cases:

<u>Case 1</u>
$A = A_1$ $B = B_i$ for some i, and we assume for simplicity i = 1, i.e. we get

$$B_2 \ldots B_m \longrightarrow A_1 \in F^+.$$

From this we can conclude that

$$F' = F \setminus \{X_1 \longrightarrow A_1\} \cup \{B_2 \ldots B_m \longrightarrow A_1\}$$

is equivalent to F, hence we get a contradiction as F was assumed to be canonical.

<u>Case 2</u>
$A = B_i$ for some i, and w.l.o.g. we put i = 1
$B = A_1$
i.e. we get

$$B_2 \ldots B_m \longrightarrow B_1 \in F^+.$$

Then consider F' as in case 1. Also in Case 2 F' is equivalent to F. To show that $F' \subset F^+$ we conclude from

$$B_2 \ldots B_m \longrightarrow B_1 \in F^+$$

that

$$B_2 \ldots B_m \longrightarrow B_1 \ldots B_m \in F^+$$

and using

$$B_1 \ldots B_m \longrightarrow A_1 \in F$$

we obtain

$$B_2 \ldots B_m \longrightarrow A_1 \in F^+.$$

To show that $F \subset (F')^+$ we conclude from

$$B_2 \dots B_m \longrightarrow A_1 \in F'$$

that

$$B_1 B_2 \dots B_m \longrightarrow B_1 A_1 \in (F')^+$$

hence

$$B_1 \dots B_m \longrightarrow A_1 \in (F')^+.$$

Hence $F' \equiv F$ and we get a contradiction because F is canonical.

Case 3

$A = B_1$, $B = B_2$ w.l.o.g.

i. e.

$$B_3 \dots B_m A_1 \longrightarrow B_1 \in F_1^+ \subseteq F^+.$$

Our first observation is that A_1 cannot be extraneous in the above FD, because otherwise $B_3 \dots B_m \longrightarrow B_1 \in F^+$ yielding a contradiction as in case 2. By Lemma 2 we conclude that there are functional dependencies g_1, g_2 in F such that

- A_1 is contained in the left-hand-side of g_1
- B_1 is the right-hand-side of g_2

and there is a directed path from g_1 to g_2 in G(R,F). By definition of G(R,F) there is an edge from f_1 to g_1 and an edge from g_2 to f_1, hence we obtain a directed cycle in G(R,F) yielding a contradiction. So (X_1A_1, F_1) is in BCNF. Obviously the same argument can be carried out for $i = 2 \dots n$.

It remains to show that the decomposition has the lossless-join property. For this let us first assume that the right-hand-sides of the FD in F are pairwise different, i.e. $A_i \neq A_j$ for $i \neq j$.

We construct ρ from R as follows:
Construct the graph G(R,F). As G(R,F) is acyclic we may choose a FD that has no outgoing edge. Let this be w.l.o.g. $f_1 = X_1 \longrightarrow A_1$. Then substitute R by

$$\rho_1 = \{S_1, S_2\}$$

where $$S_1 = R \setminus \{A_1\} \qquad S_2 = X_1 A_1,$$

then $$S_1 \cap S_2 = X_1$$

$$S_2 \setminus S_1 = A_1$$

hence $S_1 \cap S_2 \longrightarrow S_2 \setminus S_1 \in F^+$, hence by Lemma 3 i) ρ_1 is a decomposition that has the lossless-join property. We know that no $f \in F$ involves A_1 on its left-hand-side We remove f_1 from the graph G(R,F) together with all edges ending in it. We choose again a FD that has no outgoing edges, say f_2, and use f_2 to decompose S_1 as above. As G(R,F) is acyclic we end up by having taken all $f \in F$ into account.
The resulting scheme is

$$\rho = (R \setminus \{A_1, \dots A_n\}, X_1 A_1, \dots X_n A_n)$$

and has the lossless-join property.

Now let us consider the case that some FDs in F have the same right-hand-side. Let $f_i = X_i \longrightarrow A_i$ i = 1 ... r have the same right-hand-side. Then it is clear that f_1 has no outgoing edge in G(R,F) iff f_i has no outgoing edge, i = 2 ... r. Instead of substituting a scheme R by

$$(R \setminus \{A_1\}, X_1 A_1)$$

we substitute R by

$$(R \setminus \{A_1\}, X_1 A_1, X_2 A_2, \dots , X_r A_r)$$

and do the same steps later. Addition of relation schemes to a decomposition that has the lossless-join property preserves this property, so the resulting scheme

$$\rho\ (R \setminus \{A_1 \dots A_n\}, X_1 A_1, \dots , X_n A_n)$$

has the lossless-join property.

Given an arbitrary relational description (R,F) we know by Lemma 2 that any canonical cover of (R,F) can be used to detect cycles. So, we construct a canonical cover $(R,\bar{F})$ in $O(|R|^2 \cdot |F|^2)$ steps, see [12]. The size of $\bar{F}$ is $O(|R| \cdot |F|)$.

Hence the detection of cycles in $G(R,\bar{F})$ will cost $O(|R|^2 \cdot |F|^2)$ steps. The size N of the input (R,F) is $O(|R| \cdot |F|)$, so the above procedure will take cost $O(N^2)$.

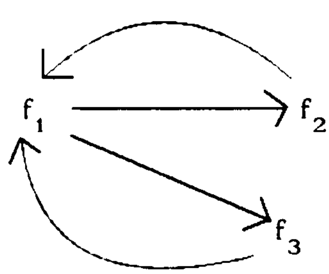
For complexity reasons it is clear that our above criterion, i.e. the cycle freeness of some canonical cover, cannot be a necessary condition for the existence of a BCNF-decomposition that preserves functional dependencies. In example 6 we exhibit a "minimal" example of a relational description that violates our criterion but does possess a BCNF-decomposition that preserves functional dependencies and has the lossless-join property.

Example 6

R = ABCD
$F = \{f_1 = AB \longrightarrow C,\ f_2 = C \longrightarrow A,\ f_3 = C \longrightarrow B\}$.
F is canonical.
Here

$$\rho = \{ABC, ABD\}$$

is a BCNF-decomposition that has the lossless-join property and preserves functional dependencies. The graph G(R,F) is



REFERENCES

1. ARMSTRONG, W.W. — Dependency structures of data base relationships. IFIP 74 (North Holland, 1974) 580 - 583

2. BEERI, C., BERNSTEIN, P.A. — Computational problems related to the design of normal form relational schemes. ACM TODS 5,3, September 1980, 241-259

3. BERNSTEIN, P.A. — Synthesizing third normal form relations from functional dependencies. ACM TODS 1:4 (Dez. 1976) 277 - 298

4. BISKUP, J., DAYAL, U., BERNSTEIN, P.A. — Synthesizing independent data base schemes, ACM SIGMOD Conf. 1979, 143-152

5. CODD, E.F. — A relational model for large shared data banks. CACM 13,6 (June 1970), 377-387

6. CODD, E.F. — Further normalization of the data base relational model. In Rustin (ed.): Courant Computer Science Symposium 6, data base systems (Prentice Hall, 1972), 33-64

7. CODD, E.F. — Recent investigations in relational data base systems. IFIP 1974 (North Holland, 1974), 1017-1021

8. DELOBEL, C., CASEY, R.G. — Decomposition of a data base and the theory of boolean switching functions. IBM J. of Res. Develop. 17 (1972) 374-386

9. FAGIN, R. — Functional dependencies in a relational data base and propositional logic. IBM Jounal of Res. Develop. 21 (1977) 534-544

10. GAREY, M.R., JOHNSON, D.S. — Computers and intractability. A guide to the theory of NP-completeness (Freeman 1979)

11. LE DOUX, C.H., PARKER, D.S. — A new algorithm for testing Boyce Codd normal form (1980). Unpublished manuscript

12. MAIER, D. — The theory of relational data bases. (Computer Science Press 1973)

13. OSBORN, S. — Testing for existence of a covering Boyce-Codd normal form. Information Proc. Letters 8,1 (1979) 11-14

14. TSOU, D.M., FISCHER, P.C. — Decomposition of a relation scheme into Boyce-Codd normal form. SIGACT NEWS 14,3 (1982) 23-29

15. ULLMANN, J.D. — Data base and knowledge base systems, Vol. I (Computer Science Press 1988)